SIEMENS

Prólogo 1

Módulo interfaz serie 2

Modbus/USS 3

SIMATIC

ET 200S
Módulos interfaz serie

Instrucciones de servicio

03/2009
A5E00259730-05

Notas jurídicas

Filosofía en la señalización de advertencias y peligros

Este manual contiene las informaciones necesarias para la seguridad personal así como para la prevención de daños materiales. Las informaciones para su seguridad personal están resaltadas con un triángulo de advertencia; las informaciones para evitar únicamente daños materiales no llevan dicho triángulo. De acuerdo al grado de peligro las consignas se representan, de mayor a menor peligro, como sigue.

⚠PELIGRO

Significa que, si no se adoptan las medidas preventivas adecuadas **se producirá** la muerte, o bien lesiones corporales graves.

⚠ADVERTENCIA

Significa que, si no se adoptan las medidas preventivas adecuadas **puede producirse** la muerte o bien lesiones corporales graves.

⚠PRECAUCIÓN

con triángulo de advertencia significa que si no se adoptan las medidas preventivas adecuadas, pueden producirse lesiones corporales.

PRECAUCIÓN

sin triángulo de advertencia significa que si no se adoptan las medidas preventivas adecuadas, pueden producirse daños materiales.

ATENCIÓN

significa que puede producirse un resultado o estado no deseado si no se respeta la consigna de seguridad correspondiente.

Si se dan varios niveles de peligro se usa siempre la consigna de seguridad más estricta en cada caso. Si en una consigna de seguridad con triángulo de advertencia se alarma de posibles daños personales, la misma consigna puede contener también una advertencia sobre posibles daños materiales.

Personal cualificado

El producto/sistema tratado en esta documentación sólo deberá ser manejado o manipulado por **personal cualificado** para la tarea encomendada y observando lo indicado en la documentación correspondiente a la misma, particularmente las consignas de seguridad y advertencias en ella incluidas. Debido a su formación y experiencia, el personal cualificado está en condiciones de reconocer riesgos resultantes del manejo o manipulación de dichos productos/sistemas y de evitar posibles peligros.

Uso previsto o de los productos de Siemens

Considere lo siguiente:

⚠ADVERTENCIA

Los productos de Siemens sólo deberán usarse para los casos de aplicación previstos en el catálogo y la documentación técnica asociada. De usarse productos y componentes de terceros, éstos deberán haber sido recomendados u homologados por Siemens. El funcionamiento correcto y seguro de los productos exige que su transporte, almacenamiento, instalación, montaje, manejo y mantenimiento hayan sido realizados de forma correcta. Es preciso respetar las condiciones ambientales permitidas. También deberán seguirse las indicaciones y advertencias que figuran en la documentación asociada.

Marcas registradas

Todos los nombres marcados con ® son marcas registradas de Siemens AG. Los restantes nombres y designaciones contenidos en el presente documento pueden ser marcas registradas cuya utilización por terceros para sus propios fines puede violar los derechos de sus titulares.

Exención de responsabilidad

Hemos comprobado la concordancia del contenido de esta publicación con el hardware y el software descritos. Sin embargo, como es imposible excluir desviaciones, no podemos hacernos responsable de la plena concordancia. El contenido de esta publicación se revisa periódicamente; si es necesario, las posibles las correcciones se incluyen en la siguiente edición.

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
ALEMANIA

A5E00259730-05
℗ 05/2011

Copyright © Siemens AG 2009.
Sujeto a cambios sin previo aviso

Índice

1 Prólogo 7

2 Módulo interfaz serie 9

2.1 Presentación del producto 9

2.2 Instrucciones breves sobre la puesta en marcha del módulo interfaz serie 13

2.3 Esquema de conexiones con asignación de pines 18

2.4 Interfaz RS-232C 24

2.5 Interfaz RS-422/485 25

2.6 Conceptos básicos de la transmisión de datos en serie 26
2.6.1 Transmisión de datos en serie 26
2.6.2 Trama de caracteres 28
2.6.3 Procedimiento de transmisión en el acoplamiento punto a punto 31
2.6.4 Seguridad de transmisión 33

2.7 Transmisión de datos con el procedimiento 3964(R) 35
2.7.1 Conceptos básicos de la transmisión de datos con el procedimiento 3964(R) 35
2.7.2 Enviar datos con el procedimiento 3964(R) 37
2.7.3 Recibir datos con el procedimiento 3964(R) 39
2.7.4 Tratamiento de errores con el procedimiento 3964(R) 41

2.8 Transmisión de datos con el driver ASCII 44
2.8.1 Conceptos básicos de la transmisión de datos con el driver ASCII 44
2.8.2 Enviar datos con el driver ASCII 44
2.8.3 Recibir datos con el driver ASCII 46
2.8.4 Criterios de fin para la transmisión de datos con el driver ASCII 48
2.8.5 Señales cualificadoras RS-232C para la transmisión de datos con el driver ASCII 51

2.9 Configuración y parametrización del módulo interfaz serie 55
2.9.1 Configuración del módulo interfaz serie 55
2.9.2 Parametrización del driver ASCII 55
2.9.3 Parametrización de los drivers para el protocolo 3964(R) 60
2.9.4 Datos de identificación 63
2.9.5 Cargar actualizaciones de firmware a posteriori 65

2.10 Comunicación a través de bloques de función 67
2.10.1 Principios de la comunicación a través de bloques de función 67
2.10.2 Bloque de función FB3 S_SEND 69
2.10.3 Bloque de función FB2 S_RCV 73
2.10.4 Funciones para parametrizar opciones para el control del flujo de datos 77
2.10.5 Lectura y forzado de señales cualificadoras de RS-232C 84

2.11 Propiedades de arranque y estados operativos 89

2.12 Datos de referencia para otros maestros como S7-PROFIBUS ... 92
2.12.1 Principios de los datos de referencia ... 92
2.12.2 Ejemplo del proceso de transmisión de datos de la CPU al módulo ... 98
2.12.3 Ejemplo de proceso al recibir datos del módulo en la CPU ... 101
2.12.4 Ejemplo de proceso al leer el estado de señalización V.24 ... 103
2.12.5 Ejemplo de proceso al escribir señales V.24 ... 104
2.12.6 Parámetros para el control del flujo de datos ... 105
2.12.7 Tratamiento de errores ... 108
2.13 Diagnóstico ... 110
2.14 Datos técnicos ... 117
3 Modbus/USS ... 121
3.1 Presentación del producto ... 121
3.2 Instrucciones breves sobre la puesta en marcha del módulo interfaz serie ... 124
3.3 Esquema de conexiones con asignación de pines ... 132
3.3.1 Asignación de pines ... 132
3.3.2 Interfaz RS-232C ... 138
3.3.3 Interfaz RS-422/485 ... 141
3.4 Protocolo de transmisión Modbus ... 142
3.4.1 Propiedades y estructura del telegrama ... 142
3.4.2 Dirección de esclavo ... 143
3.4.3 Códigos de función de maestro y esclavo ... 144
3.4.4 Campo de datos DATA ... 144
3.4.5 Fin de telegrama y comprobación CRC ... 145
3.4.6 Respuestas de excepción ... 146
3.5 Driver maestro Modbus ... 148
3.5.1 Empleo del driver maestro Modbus ... 148
3.5.2 Transferencia de datos con el maestro de Modbus ET 200S ... 149
3.5.3 Configuración y parametrización del maestro Modbus ... 158
3.5.4 Códigos de función empleados por el maestro Modbus ... 163
3.5.5 Código de función 01 – Read Output Status ... 164
3.5.6 Código de función 02 – Read Input Status ... 165
3.5.7 Código de función 03 – Read Output Registers ... 166
3.5.8 Código de función 04 – Read Input Registers ... 167
3.5.9 Código de función 05 -- Force Single Coil ... 168
3.5.10 Código de función 06 – Preset Single Register ... 169
3.5.11 Código de función 07 – Read Exception Status ... 170
3.5.12 Código de función 08 -- Loop Back Diagnostic Test ... 171
3.5.13 Código de función 11 – Fetch Communications Event Counter ... 172
3.5.14 Código de función 12 – Fetch Communications Event Log ... 173
3.5.15 Código de función 15 – Force Multiple Coils ... 174
3.5.16 Código de función 16 – Preset Multiple Registers ... 175

3.6 Driver esclavo de Modbus 176
3.6.1 Componentes del acoplamiento de esclavo Modbus 176
3.6.2 Transferencia de datos con el esclavo Modbus ET 200S 178
3.6.3 Áreas de datos en la CPU SIMATIC 180
3.6.4 Configurar los parámetros para el acoplamiento 182
3.6.5 Códigos de función del esclavo 187
3.6.6 Código de función 01 – Read Coil (Output) Status 188
3.6.7 Código de función 02 – Read Input Status 192
3.6.8 Código de función 03 – Read Output Registers 195
3.6.9 Código de función 04 – Read Input Registers 198
3.6.10 Código de función 05 – Force Single Coil 201
3.6.11 Código de función 06 – Preset Single Register 204
3.6.12 Código de función 08 – Loop Back Diagnostic Test 207
3.6.13 Código de función 15 – Force Multiple Coils 208
3.6.14 Código de función 16 – Preset Multiple Registers 211
3.6.15 Conversión del código de función orientada al bit 214
3.6.16 Conversión de códigos de función orientada en fichas 215
3.6.17 Habilitar/bloquear accesos de escritura 216
3.6.18 Conversión de las direcciones de Modbus para funciones de bit 217
3.6.19 Conversión de las direcciones de MODBUS para funciones de ficha 222
3.6.20 Límites para funciones de escritura 225

3.7 Diagnóstico 227
3.7.1 Posibilidades de diagnóstico 227
3.7.2 Información de diagnóstico de los LEDs de estado 227
3.7.3 Avisos de diagnóstico de los bloques de función 228
3.7.4 Diagnóstico de esclavos PROFIBUS 238
3.7.5 Funciones de diagnóstico del esclavo Modbus 239
3.7.6 Errores 240

3.8 Maestro de USS 243
3.8.1 ¿Qué es el maestro de USS? 243
3.8.2 Protocolo de USS 244
3.8.3 Configuración y parametrización 246
3.8.4 Resumen de funciones 247
3.8.5 FC17 S_USST: Enviar datos a un esclavo 249
3.8.6 FC18 S_USSR: Recibir datos de un esclavo 252
3.8.7 FC19 S_USSI: Inicialización 256
3.8.8 DB de datos de red 259
3.8.9 DB de parametrización 265
3.8.10 DB del procesador de comunicaciones 267

3.9 Propiedades de arranque y estados operativos del módulo interfaz serie ET 200S Modbus/USS 269
3.9.1 Carga de los datos de configuración y parametrización 269
3.9.2 Estados operativos del módulo interfaz serie ET 200S Modbus/USS 270
3.9.3 Propiedades de arranque del módulo interfaz serie ET 200S Modbus/USS 271
3.9.4 Comportamiento del módulo interfaz serie ET 200S Modbus/USS en caso de cambio de estado operativo de la CPU 271

3.10 Datos técnicos 273

Índice alfabético 279

Prólogo

1

Estructura del manual

Este manual es un suplemento del manual *Sistema de periferia descentralizada ET 200S*.

En el manual *Sistema de periferia descentralizada ET 200S* encontrará información detallada sobre la configuración de hardware, la estructura, el cableado, la puesta en marcha, el diagnóstico y los datos técnicos del sistema de periferia descentralizada ET 200S.

En el presente manual encontrará la descripción de las funciones y los datos técnicos de los módulos interfaz serie ET 200S 1SI y ET 200S Modbus/USS.

Orientación

Al principio de cada capítulo hay una **Presentación del producto** en la que se listan las propiedades del módulo descrito y también las diferentes posibilidades de empleo. En ella encontrará además una indicación a la referencia del módulo descrito y al nombre y la versión del software. El archivo GSD actual está disponible en la siguiente dirección de Internet:

http://support.automation.siemens.com

A continuación encontrará en cada capítulo una **breve introducción a la puesta en marcha**. En estas instrucciones breves se explica en pequeños pasos cómo montar, configurar, integrar en el programa de usuario y probar el módulo correspondiente.

Normas y homologaciones

Para más información sobre las normas y homologaciones que cumple este sistema, consulte el capítulo "Especificaciones técnicas generales" del manual *Sistema de periferia descentralizada ET 200S*. Encontrará dicho manual en:

http://www.siemens.com/simatic-tech-doku-portal

Reciclaje y gestión de residuos

Los módulos interfaz serie ET 200S 1SI 3964/ASCII y ET 200S 1SI Modbus/USS son reciclables gracias a su reducida composición de materias nocivas. Para un reciclaje y una eliminación ecológica de los equipos usados, rogamos dirigirse a un centro certificado de recogida de material electrónico.

Índice

El índice contiene palabras clave referentes a todos los capítulos.

Soporte técnico

Se puede acceder al servicio Technical Support para todos los productos Industry Automation utilizando el formulario online para Support Request que encontrará en http://www.siemens.com/automation/support-request

Encontrará más información sobre nuestro Technical Support en la dirección de Internet http://www.siemens.com/automation/service

Service & Support en Internet

Además de la documentación disponible, en Internet le ofrecemos todo nuestro know-how.

http://www.siemens.com/automation/service&support

Allí encontrará:

- el boletín de novedades, que le mantendrá completamente informado sobre las actualizaciones de sus productos.
- los documentos que necesita con el buscador del Product Support.
- un foro en el que intercambian experiencias usuarios y especialistas de todo el mundo.
- El especialista o experto de Industry de su región.
- Información sobre reparaciones, recambios y consultoría.

Asesoramiento adicional

Si tiene dudas sobre la utilización de los productos descritos en el manual que éste no resuelva, diríjase a los representantes de Siemens en las instalaciones que le correspondan.

Encontrará a su representante en:

http://www.siemens.com/automation/partner

Puede encontrar la guía de orientación sobre la oferta de documentaciones técnicas para los productos y sistemas individuales de SIMATIC en:

http://www.siemens.de/simatic-tech-doku-portal

Encontrará el catálogo online y el sistema de pedidos online en:

http://mall.automation.siemens.com

Centro de formación

Para facilitar la iniciación a los sistemas de automatización SIMATIC S7, ofrecemos distintos cursillos de formación. Le rogamos se dirija a su centro de formación regional o al centro de formación central en D 90327 Nürnberg.

http://www.sitrain.com

Módulo interfaz serie

2

2.1 Presentación del producto

Número de referencia

6ES7 138-4DF01-0AB0

Descripción del producto

El módulo interfaz serie ET 200S 1SI es un módulo enchufable de la serie de productos ET 200S. Permite acceder a la comunicación serie a través de tres interfaces de hardware (RS232C, RS-422 y RS485) y dos protocolos de software (ASCII y 3964(R)).

Con el módulo interfaz ET 200S 1SI es posible intercambiar datos entre sistemas de automatización u ordenadores a través de un acoplamiento punto a punto. Toda la comunicación se realiza mediante transferencias asíncronas en serie.

El tipo de comunicación se elige al parametrizar el módulo en la configuración de hardware de STEP 7 o en otra aplicación de configuración. En el catálogo de hardware aparecen los seis modelos siguientes del módulo:

- ASCII (4B)
- ASCII (8B)
- ASCII (32B)
- 3964R (4B)
- 3964R (8B)
- 3964R (32B)

Las transmisiones de datos de 8 ó 32 bytes aumentan el caudal pero requieren más memoria E/S en el bastidor ET 200S, mientras que las transmisiones de datos de 4 bytes requieren menos memoria E/S en el bastidor ET 200S, pero ofrecen un caudal menor. La variante del módulo depende de los requisitos de cada aplicación.

Funciones del módulo interfaz serie ET 200S 1SI

El módulo interfaz serie ET 200S 1SI ofrece las funciones siguientes:

- Interfaz integrada según RS232C, RS-422 ó RS-485
- Velocidad de transmisión hasta 115,2 Kbaudios, semidúplex
- Integración de los siguientes protocolos de transmisión en el firmware del módulo:
  - Procedimiento 3964(R)
  - Driver ASCII

La parametrización del módulo determina las funciones de los drivers.

La tabla siguiente lista las funciones de las diferentes interfaces del driver.

Tabla 2- 1 Funciones de los drivers de módulo del módulo interfaz serie ET 200S 1SI

Función	RS-232C	RS-422	RS-485
Driver ASCII	**Sí**	**Sí**	**Sí**
Utilización de señales cualificadoras de RS232C	Sí	No	No
Forzado/lectura de señales cualificadoras de RS232C con FBs	Sí	No	No
Control de flujo con RTS/CTS	Sí	No	No
Control de flujo con XON/XOFF	Sí	Sí	No
Procedimiento 3964(R)	**Sí**	**Sí**	**No**

Comunicación

El módulo interfaz serie ET 200S 1SI permite un acoplamiento punto a punto con diferentes módulos de Siemens y también con productos de terceros, entre ellos:

- SIMATIC S5 mediante el driver 3964(R) con el correspondiente módulo interfaz en el lado de S5
- Terminales SIEMENS BDE familia ES 2 vía driver 3964(R)
- MOBY I (ASM 420/421, SIM), MOBY L (ASM 520) y estación de registro ES 030K vía driver 3964(R)
- SIMOVERT y SIMOREG (protocolo USS) mediante el driver ASCII (ET 200S SI RS 422/485) con la correspondiente adaptación de protocolo con un programa STEP 7
- PCs mediante procedimiento 3964(R) (existen herramientas de desarrollo para programación en PC: PRODAVE DOS 64R (6ES5 897–2UD11) para MS-DOS, PRODAVE WIN 64R (6ES5 897-2VD01) para Windows o el driver ASCII)
- Lector de código de barras vía driver 3964(R) o driver ASCII
- Autómatas programables de otros fabricantes a través del driver 3964(R) o el driver ASCII
- Otros aparatos con estructuras de protocolo sencillas, mediante la correspondiente adaptación del protocolo con el driver ASCII
- Otros equipos que también dispongan de driver 3964(R)

Nota

¡Detrás de los CPs de comunicación externos CP342-5 (Profibus DP) y CP343-1 (Profinet IO) no puede emplearse el módulo ET 200S 1SI con los FBs estándar normales!

Para el uso del módulo después del CP de comunicaciones CP 342-5 (Profibus DP) o CP 343-1 (Profinet IO) existen FBs especiales disponibles en las páginas de Internet de Customer Support:

Véase http://support.automation.siemens.com/WW/view/com/26263724

Indicadores LED

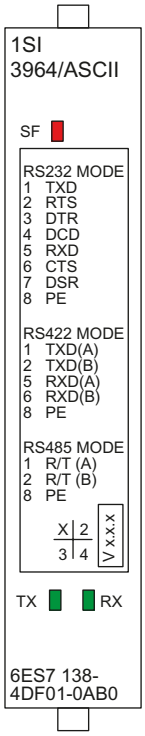
Los siguientes indicadores LED se encuentran en el panel frontal del módulo interfaz:

LED	Color	Descripción
SF	Rojo	Indicador de error colectivo
TX	Verde	La interfaz está enviando.
RX	Verde	La interfaz está recibiendo.

Los estados operativos y los errores indicados por estos LEDs se describen en el apartado Diagnóstico (Página 110).

Panel frontal

La figura muestra la rotulación del panel frontal del módulo interfaz serie ET 200S 1SI.



2.2 Instrucciones breves sobre la puesta en marcha del módulo interfaz serie

Introducción

En estas instrucciones breves se muestra un ejemplo sobre el envío y la recepción de datos entre módulos interfaz serie, que sirve de base para explicar cómo se crea una aplicación operativa, cómo funcionan las operaciones básicas del módulo interfaz serie (hardware y software) y cómo se verifican el hardware y el software.

En este ejemplo se utilizan dos módulos interfaz serie ET 200S 1SI en modo RS-232C ASCII.

Requisitos

Deben cumplirse los requisitos siguientes:

- Se pone en marcha una estación ET 200S en un equipo S7 con maestro DP.
- Se necesitan los componentes siguientes:
  - Dos módulos terminal TM-E15S24-01
  - Dos módulos interfaz serie ET 200S 1SI 3964(R)/ASCII
  - El material de cableado necesario

Montar, cablear y equipar

Monte y efectúe el cableado de ambos módulos de terminal TM-E15S24-01 (véase la siguiente figura). Conecte los dos módulos interfaz serie ET 200S 1SI 3964(R)/ASCII con los módulos de terminal. (Encontrará información detallada en el manual *Sistema de periferia descentralizada ET 200S*).

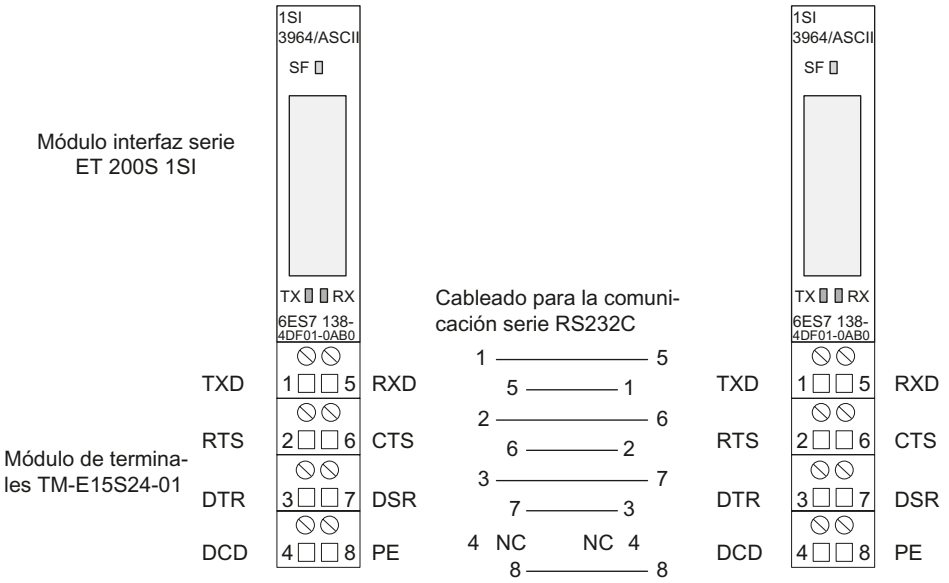


Figura 2-1 Asignación de pines del ejemplo

Configuración utilizada

En la siguiente tabla encontrará la configuración utilizada para el programa de ejemplo.

Tabla 2- 2 Parametrización de la aplicación de ejemplo

Parámetro	Valor
Diagnóstico colectivo	Bloquear
Interfaz	RS232C
Línea de recepción preasignada	Irrelevante para RS232
Control del flujo de datos (preajuste)	Ninguno
Velocidad de transmisión	9600
Bits de datos	8
Bits de parada	1
Paridad	Par
Recepción de carácter de fin	Transcurrido el tiempo de retardo entre caracteres
Tiempo de retardo entre caracteres (ms)	4
Carácter de fin 1	Irrelevante para RS232

Parámetro	Valor
Carácter de fin 2	Irrelevante para RS232
Recepción de cantidad de caracteres	Irrelevante para RS232
Búfer dinámico de telegramas	Sí
No sobrescribir búfer de telegramas	Sí
Borrar el búfer de recepción durante el arranque	Sí

Bloques utilizados

En la siguiente tabla encontrará los bloques utilizados en el programa de ejemplo.

Bloque	Símbolo	Comentario
OB 1	CYCLE	Ejecución cíclica del programa
OB 100	RESTART	Ejecución de rearranque completo
DB 21	SEND_IDB_SI_0	DB de instancia para FB S_SEND_SI
DB 22	RECV_IDB_SI_1	DB de instancia para FB S_RECV_SI
DB 40	SEND_WORK_DB_SI_0	DB de trabajo para FB estándar 3
DB 41	RECV_WORK_DB_SI_1	DB de trabajo para FB estándar 2
DB 42	SEND_SRC_DB_SI_0	Bloque de datos de envío
DB 43	RECV_DST_DB_SI_1	Bloque de datos de recepción
FB 2	S_RECV_SI	FB estándar para recibir datos
FB 3	S_SEND_SI	FB estándar para enviar datos
FC 21	SEND_SI_0	Enviar datos
FC 22	RECV_SI_1	Recibir datos

Suministro e instalación

El programa de ejemplo del módulo ET 200S 1SI está disponible, junto con los bloques de función, en la dirección de Internet siguiente:

http://support.automation.siemens.com/WW/view/es/10805265/133100

Después de la instalación, el programa de ejemplo se encuentra en el proyecto zXX21_10_1SI_ASCII.

El proyecto se abre en el SIMATIC Manager de STEP 7 con el comando de menú "Archivo > Abrir > Proyectos de ejemplo".

El ejemplo de programación está disponible de forma compilada y en forma de archivo fuente ASCII. Asimismo, hay una tabla de símbolos con los símbolos empleados en el ejemplo.

Si no se dispone de un segundo ET 200S 1SI como interlocutor, hay que eliminar el segundo ET 200S 1SI en HW Config con el comando "Edición > Borrar". Adicionalmente, en el OB 1 debe quitarse el comentario de la llamada de la FC 22 (FC para Receave).

Cargar en la CPU

El hardware para el ejemplo está completamente montado y la unidad de programación está conectada.

Tras el borrado total de la CPU (estado operativo STOP), transfiera el ejemplo completo a la memoria de usuario. A continuación conmute el interruptor de modo de operación de STOP a RUN.

Comportamiento erróneo

Si durante el arranque se produce un error, los accesos cíclicos a los módulos no se llevan a cabo, y se activa la indicación de error.

Si se produce un mensaje de error, se activa el parámetro de salida ERROR de los bloques. En el parámetro STATUS de los bloques hay una descripción más detallada de los errores. Si el mensaje de error 16#1E0E o 16#1E0F se halla en STATUS, la descripción detallada del error se encuentra almacenada en las variables SFCERR del DB de instancia.

Conexión, programa de arranque

El programa de arranque se halla en el OB 100.

En el arranque se desactivan los bits de control y los contadores

Programa cíclico

El programa cíclico se halla en el OB 1.

En el ejemplo, los bloques de función FB 2 S_RECV_SI y FB 3 S_SEND_SI operan conjuntamente con las funciones FC 21 y FC 22, así como con los bloques de datos DB 21 y DB 22 como DBs de instancia y con DB 42 y DB 43 como DB emisor o receptor.

La parametrización de los bloques de función de este ejemplo se produce en parte con constantes y en parte con operandos actuales direccionados simbólicamente.

Descripción

La transmisión de datos tiene lugar desde el ET 200S 1SI, insertado en el slot 2, al ET 200S 1SI, insertado en el slot 3. Si se trabaja con otro interlocutor, no se produce la llamada de la FC 22 (RECEIVE).

Descripción de la FC 21 (SEND)

Sección del programa "Generate edge S_SEND_SI_REQ":

S_SEND_SI se ejecuta una vez al comienzo con S_SEND_SI_ REQ=0. A continuación, S_SEND_SI_ REQ recibe el valor 1. Si en el parámetro de control S_SEND_SI_ REQ se detecta un cambio de estado de señal de 0 a 1, se inicia la petición S_SEND_SI.

Con S_SEND_SI_ DONE=1 ó S_SEND_SI_ERROR=1, S_SEND_SI_REQ vuelve a adquirir del valor 0.

Sección del programa "S_SEND_SI_DONE=1":

Si la transferencia finaliza con éxito, en la salida de parámetro de S_SEND_SI, el parámetro S_SEND_SI_DONE recibe el valor 1.

A fin de poder distinguir transferencias sucesivas, en la palabra de datos 0 del bloque fuente DB 42 se añade un contador de transferencias S_SEND_COUNTER_OK.

Sección del programa "S_SEND_SI_ERROR=1":

Si S_SEND_SI se ejecuta con S_SEND_SI_ERROR=1, en la palabra de datos 2 el contador de errores S_SEND_SI_COUNTER_ERR incrementa. Además, se hace una copia de S_SEND_SI_WORK_STAT, ya que en el próximo acceso será sobrescrito con 0 y ya no podrá consultarse.

Descripción de la FC 22 (RECEIVE)

Sección del programa "Enable Receive Data":

Para recibir datos es necesario que la habilitación de recepción S_RECV_SI_EN_R del bloque S_RECV_SI tenga el valor 1.

Sección del programa "S_RECV_SI_NDR=1":

Si S_RECV_SI_NDR está activado, significa que se han recibido datos nuevos y el contador de recepción S_RECV_SI_WORK_CNT_OK incrementa.

Sección del programa "S_RECV_SI_ ERROR=1":

En caso de error durante el arranque, es decir, si el bit de error de la salida del parámetro S_RECV_SI está activado, el contador S_RECV_SI_WORK_CNT_ERR se incrementa. Además, se hace una copia de S_RECV_SI_WORK_STAT, ya que en el próximo acceso será sobrescrito con 0 y ya no podrá consultarse.

Todos los valores relevantes pueden consultarse en la tabla de variables con fines de comprobación.

2.3 Esquema de conexiones con asignación de pines

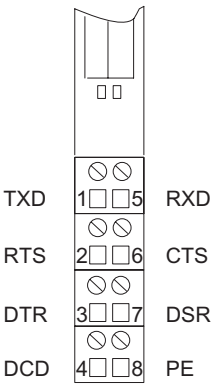
Reglas de cableado

Los cables (bornes 1 - 8) deben estar apantallados. La pantalla tiene que hacer contacto por ambos extremos. Para ello se debe usar el elemento de contacto de pantalla (véase el manual *Sistema de periferia descentralizada ET 200S 1SI*).

Asignación de pines para la comunicación RS232-C

La tabla siguiente muestra la asignación de pines para el módulo interfaz serie ET 200S 1SI con protocolo de comunicación RS232C ajustado.

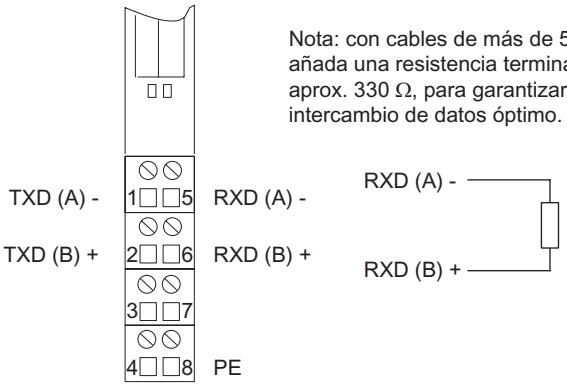
Tabla 2- 3 Asignación de pines del módulo interfaz serie ET 200S 1SI para la comunicación RS-232C

Vista	Observaciones		
	Modo: Semidúplex y dúplex		
	Bornes		
	1	TXD	Datos enviados
	5	RXD	Datos recibidos
	2	RTS	Petición de envío
	6	CTS	Listo para enviar
	3	DTR	Terminal de datos listo
	7	DSR	Registro listo
	4	DCD	Detección del soporte de datos
	8	PE	Tierra

Asignación de pines para la comunicación RS422

La tabla siguiente muestra la asignación de pines para el módulo interfaz serie ET 200S 1SI con protocolo de comunicación RS422 ajustado.

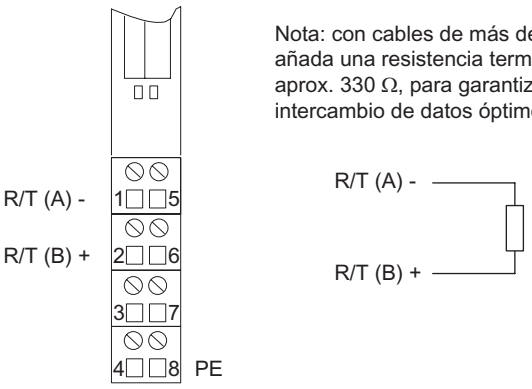
Tabla 2- 4 Asignación de pines del módulo interfaz serie ET 200S 1SI para la comunicación RS422

Vista	Asignación de pines	Observaciones	
		Modo: Dúplex Bornes	
		1	TXD (A)-
		5	RXD (A)-
		2	TXD (B)+
		6	RXD (B)+
		8	PE tierra

Asignación de pines para la comunicación RS485

La tabla siguiente muestra la asignación de pines para el módulo interfaz serie ET 200S 1SI con protocolo de comunicación RS485 ajustado.

Tabla 2- 5 Asignación de pines del módulo interfaz serie ET 200S 1SI para la comunicación RS485

Vista	Asignación de pines	Observaciones	
		Modo: Semidúplex Bornes	
		1	R/T (A)-
		2	R/T (B)+
		8	PE tierra

Asignación de pines del cable de conexión RS232C para conector macho de 9 polos

La siguiente figura muestra las conexiones de cables para la comunicación punto a punto según RS232C entre el módulo interfaz serie ET 200S 1SI y un interlocutor con un conector hembra D de 9 polos.

- En el lado del ET 200S 1SI se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el interlocutor un conector hembra SubD de 9 polos.

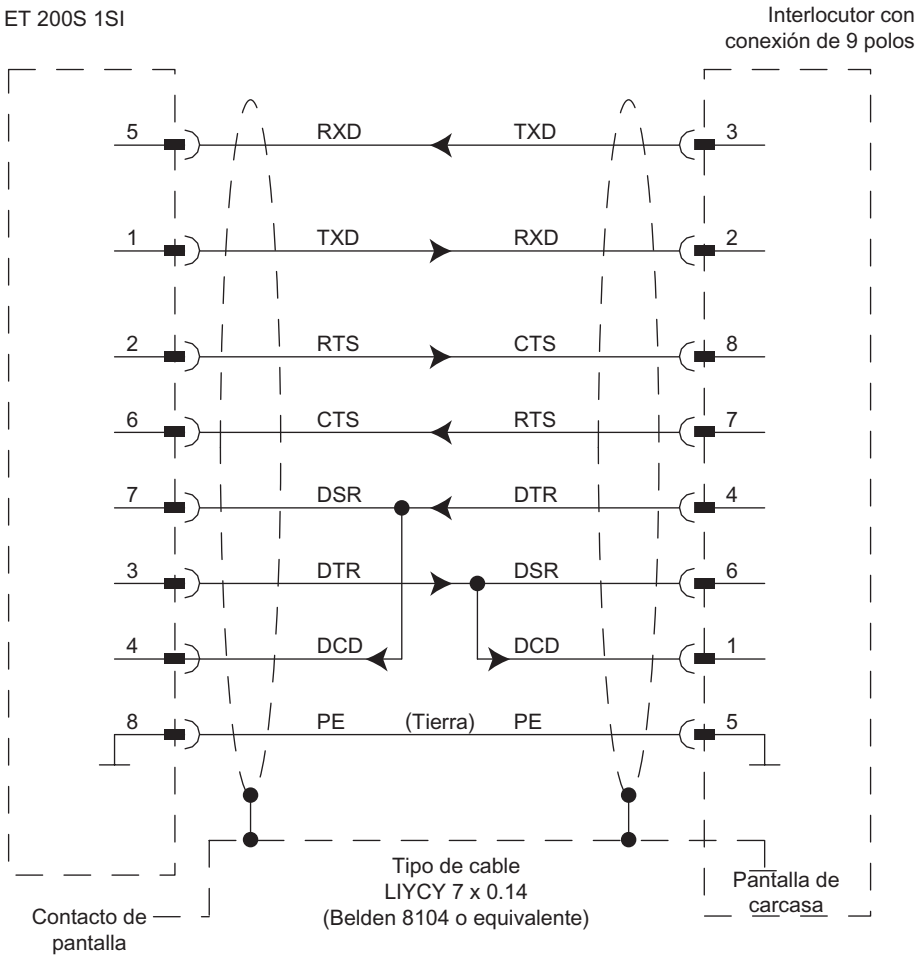


Figura 2-2 Asignación de pines del cable de conexión RS232C para conector macho de 9 polos

Asignación de pines del cable de conexión RS232C para conector macho de 25 polos

La siguiente figura muestra las conexiones de cables para la comunicación punto a punto según RS232C entre el módulo interfaz serie ET 200S 1SI y un interlocutor con un conector macho D de 25 polos.

- En el lado del ET 200S 1SI se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el interlocutor un conector macho SubD de 25 polos.

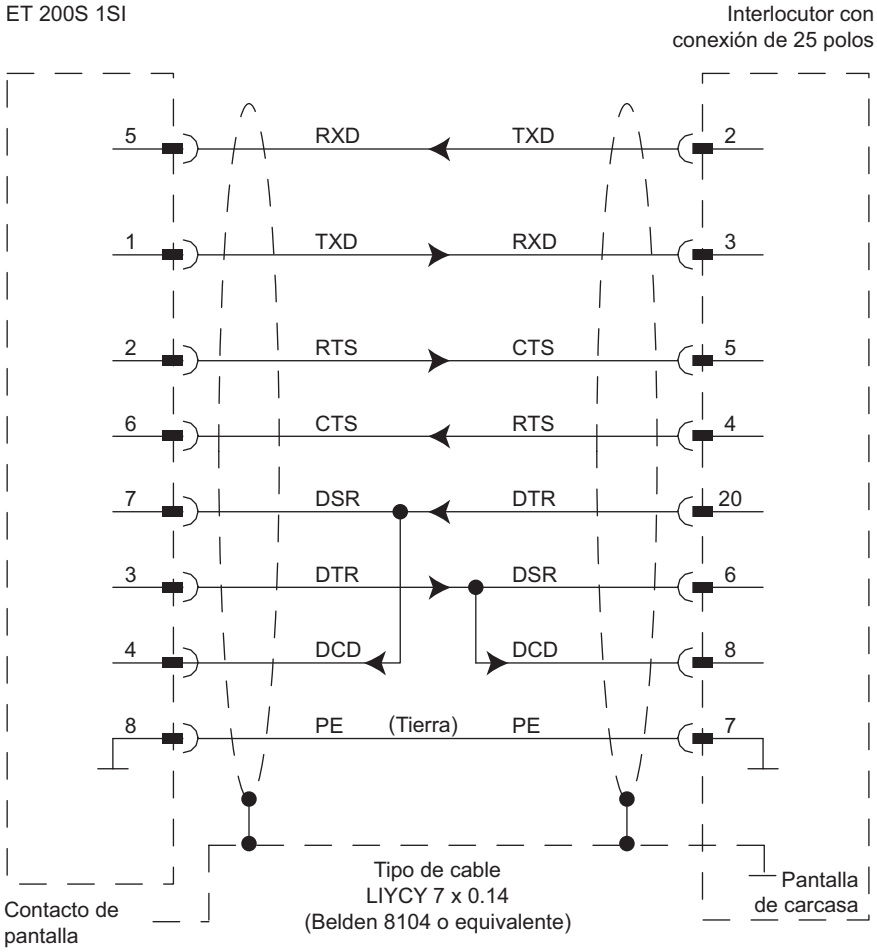


Figura 2-3 Asignación de pines del cable de conexión RS232C para conector macho de 25 polos

Asignación de terminales del cable de conexión RS422 para conectores macho de 15 polos

La siguiente figura muestra las conexiones de cables para la comunicación punto a punto según RS422 entre el módulo interfaz serie ET 200S 1SI y un interlocutor con un conector macho D de 15 polos.

- En el lado del ET 200S 1SI se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el interlocutor un conector macho SubD de 15 polos.

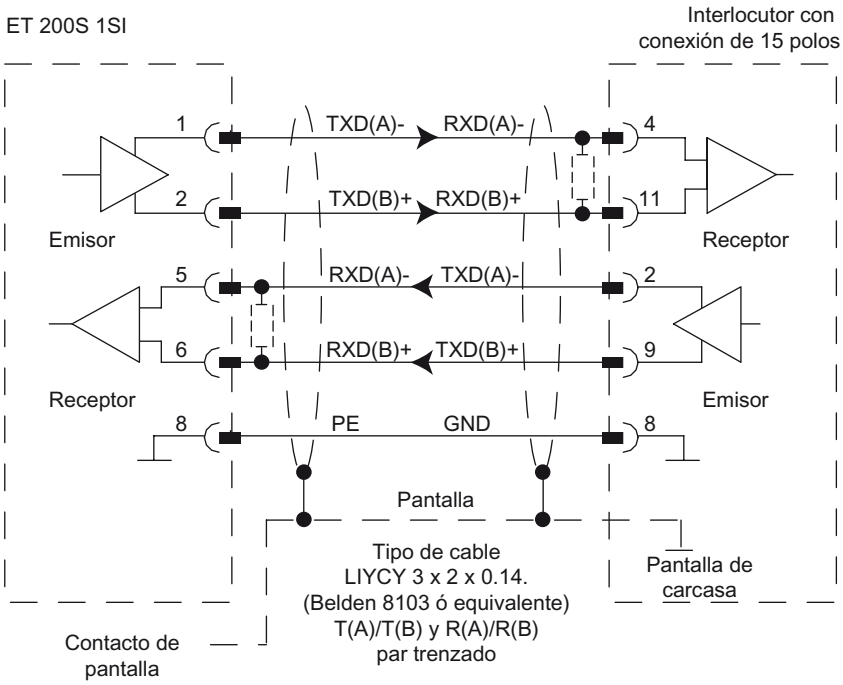


Figura 2-4 Asignación de terminales del cable de conexión RS422 para conectores macho de 15 polos

Nota

Con cables de más de 50 m añada una resistencia terminadora de aprox. 330 Ω (véase la figura superior), para garantizar un intercambio de datos óptimo.

Para el tipo de cable utilizado, puede utilizar las siguientes longitudes para el módulo ET 200S 1SI como interlocutor:

- máx. 1200 m a 19.200 baudios
- máx. 500 m a 38.400 baudios
- máx. 250 m a 76.800 baudios

Asignación de terminales del cable de conexión RS485 para conectores macho de 15 polos

La siguiente figura muestra las conexiones de cables para la comunicación punto a punto según RS485 entre el módulo interfaz serie ET 200S 1SI y un interlocutor con un conector macho D de 15 polos.

- En el lado del ET 200S 1SI se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el interlocutor un conector macho SubD de 15 polos.

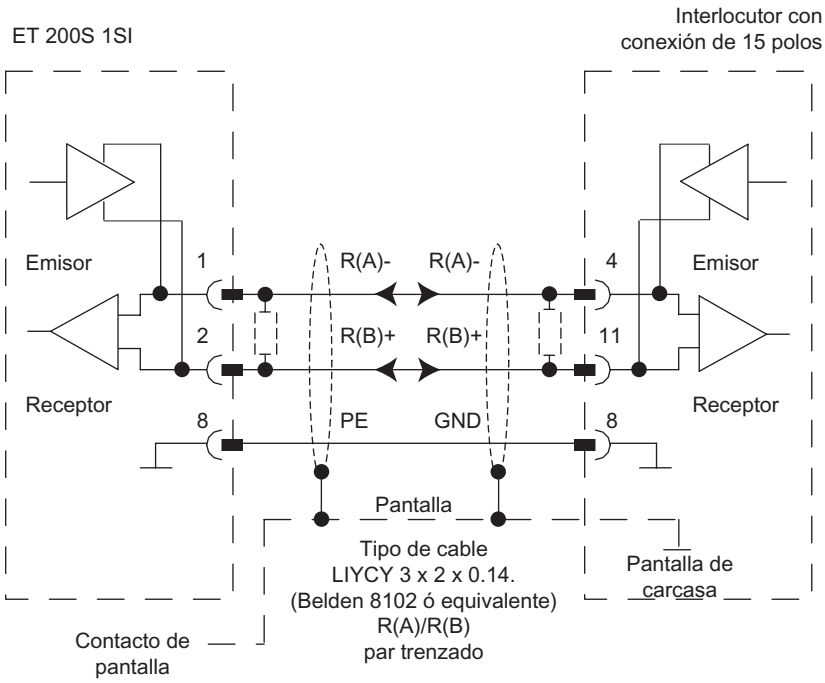


Figura 2-5 Asignación de terminales del cable de conexión RS485 para conectores macho de 15 polos

Nota

Con cables de más de 50 m añada una resistencia terminadora de aprox. 330 Ω (véase la figura superior), para garantizar un intercambio de datos óptimo.

Para el tipo de cable utilizado, puede utilizar las siguientes longitudes para el módulo ET 200S 1SI como interlocutor:

- máx. 1200 m a 19.200 baudios
- máx. 500 m a 38.400 baudios
- máx. 250 m a 76.800 baudios
- máx. 200 m a 115.200 baudios

2.4 Interfaz RS-232C

Definición

La interfaz RS 232C es una interfaz de tensión cuya función es la transmisión serie de datos de acuerdo con la norma RS 232C.

Propiedades

La interfaz RS232C tiene las propiedades siguientes:

Clase:	Interfaz de tensión
Conector frontal:	Conector de bornes estándar de 8 polos del ET 200S
Señales RS 232C:	TXD, RXD, RTS, CTS, DTR, DSR, DCD, GND
Velocidad de transmisión:	Máximo 115,2 Kbaudios (procedimiento 3964(R)) máximo 115,2 Kbaudios (driver ASCII)
Longitud de cable:	Máximo 15 m, tipo de cable LIYCY 7 x 0.14
Normas:	DIN 66020, DIN 66259, EIA-RS 232C, CCITT V.24/V.28
Grado de protección:	IP 20

Señales RS 232C

La tabla siguiente describe las señales RS232C.

Señal	Designación	Significado
TXD	**T**ransmitted **D**ata	Datos de envío; en estado de reposo la línea de envío se mantiene en "1" lógico.
RXD	**R**eceived **D**ata	Datos de recepción; el interlocutor debe mantener la línea de recepción en "1" lógico.
RTS	**R**equest **T**o **S**end	ON: ET 200S 1SI listo para enviar. OFF: ET 200S 1SI no envía.
CTS	**C**lear **T**o **S**end	El interlocutor puede recibir datos del ET 200S. El módulo interfaz espera esta señal como respuesta a RTS = ON.
DTR	**D**ata **T**erminal **R**eady	ON: ET 200S SI está conectado y listo para funcionar. OFF: ET 200S SI no está conectado ni listo para funcionar.
DSR	**D**ata **S**et **R**eady	ON: El interlocutor está conectado y listo. OFF: El interlocutor no está ni conectado ni listo.
DCD	**D**ata **C**arrier **D**etect	Señal portadora en caso de que se conecte un módem.

2.5 Interfaz RS-422/485

Definición

La interfaz RS-422/485 es una interfaz diferencial y sirve para transmitir los datos en serie según la norma RS422/485.

Propiedades

La interfaz RS422/485 tiene las propiedades siguientes:

Clase:	Interfaz de tensión diferencial
Conector frontal:	Conector de bornes estándar de 8 polos del ET 200S
Señales RS-422:	TXD (A)-, RXD (A)-, TXD (B)+, RXD (B)+, GND
Señales RS-485:	R/T (A)-, R/T (B)+, GND
Velocidad de transmisión:	Máximo 115,2 Kbaudios (procedimiento 3964(R)) máximo 115,2 Kbaudios (driver ASCII)
Longitud de cable:	Máximo 1200 m, tipo de cable LIYCY 7 x 0.14
Normas:	EIA RS-422/485, CCITT V.11/V.27
Grado de protección:	IP 20

2.6 Conceptos básicos de la transmisión de datos en serie

2.6.1 Transmisión de datos en serie

Acoplamiento punto a punto

Existen varias posibilidades de interconexión en red para intercambiar datos entre dos o más interlocutores. El acoplamiento punto a punto entre dos interlocutores es el método más sencillo de intercambio de información.

En el acoplamiento punto a punto, el módulo interfaz serie es la interfaz entre un autómata programable y un interlocutor. En el acoplamiento punto a punto, la transmisión de datos se realiza en serie con el módulo interfaz serie ET 200S 1SI.

Transmisión serie de datos

En la transmisión serie de datos, cada uno de los bits que componen un byte de la información transferida se envía sucesivamente en un orden determinado.

La transmisión de datos con el interlocutor se realiza independientemente mediante el módulo interfaz ET 200S 1SI a través de la interfaz serie. Para ello, el módulo está equipado con dos drivers diferentes para un intercambio de datos bidireccional.

- Driver ASCII
- Procedimiento 3964(R)

Intercambio de datos bidireccional - Modos de operación

En el intercambio de datos bidireccional se distinguen dos modos de operación con el ET 200S 1SI:

- Modo semidúplex (procedimiento 3964(R), driver ASCII)

  Los datos se transmiten alternativamente en ambas direcciones entre los interlocutores. Semidúplex significa que en un mismo momento sólo se envía o sólo se recibe. La excepción pueden ser algunos caracteres de control individuales para el control de flujo de datos (p. ej. XON/XOFF), que también pueden recibirse/enviarse durante el modo de envío/recepción.

- Modo dúplex (driver ASCII)

  Los datos se transfieren simultáneamente en ambos sentidos entre los interlocutores. El modo dúplex significa que puede enviarse y recibirse al mismo tiempo. Cada interlocutor debe ser capaz de controlar silmultáneamente una dirección de envío y otra de recepción.

La tabla siguiente lista los modos de operación del intercambio de datos para los tipos de interfaces con drivers ASCII.

Tabla 2- 6 Modos de operación del intercambio de datos para el módulo interfaz serie ET 200S 1SI

Transferencia de datos	RS-232C	RS-422	RS-485
Semidúplex	Sí	Sí	Sí
Dúplex	Sí	Sí	No es posible

Convenciones

Para la transmisión de datos en serie se requieren acuerdos entre ambos interlocutores. Entre ellas se incluyen:

- Velocidad de transmisión (baudrate)
- Tiempo de retardo entre caracteres y de acuse
- Paridad
- Cantidad de bits de datos
- Número de bits de parada
- Número de intentos de configuración y de transferencia

Los apartados Conceptos básicos de la transmisión de datos con el procedimiento 3964(R) (Página 35) y Conceptos básicos de la transmisión de datos con el driver ASCII (Página 44) describen la función que desempeñan los acuerdos en los diferentes procesos de transferencia así como su parametrización.

2.6.2 Trama de caracteres

Principio

Los datos entre el módulo interfaz serie ET 200S 1SI y un interlocutor se transfieren en una trama de caracteres de 10 u 11 bits a través de la interfaz serie. Para cada trama de caracteres se dispone de tres formatos de datos. El formato necesario puede parametrizarse en STEP 7.

Trama de caracteres de 10 bits

En la figura siguiente se muestran los tres formatos de datos de la trama de caracteres de 10 bits.

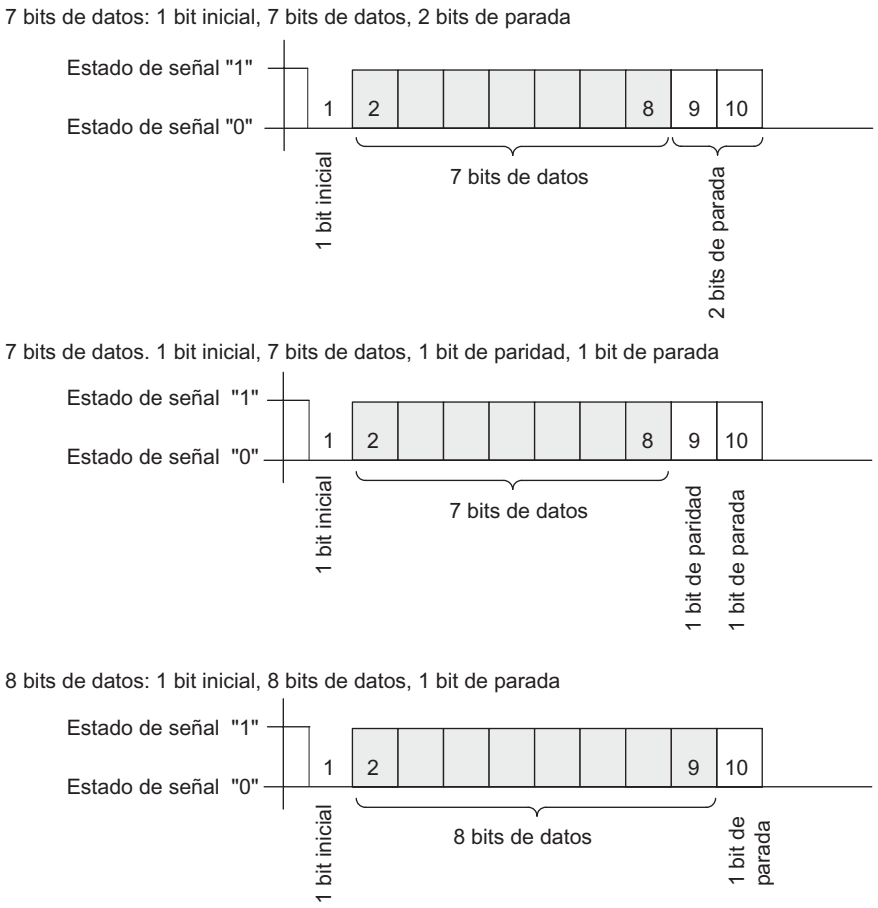


Figura 2-6 Trama de caracteres de 10 bits

Trama de caracteres de 11 bits

En la figura siguiente se muestran los tres formatos de datos de la trama de caracteres de 11 bits.

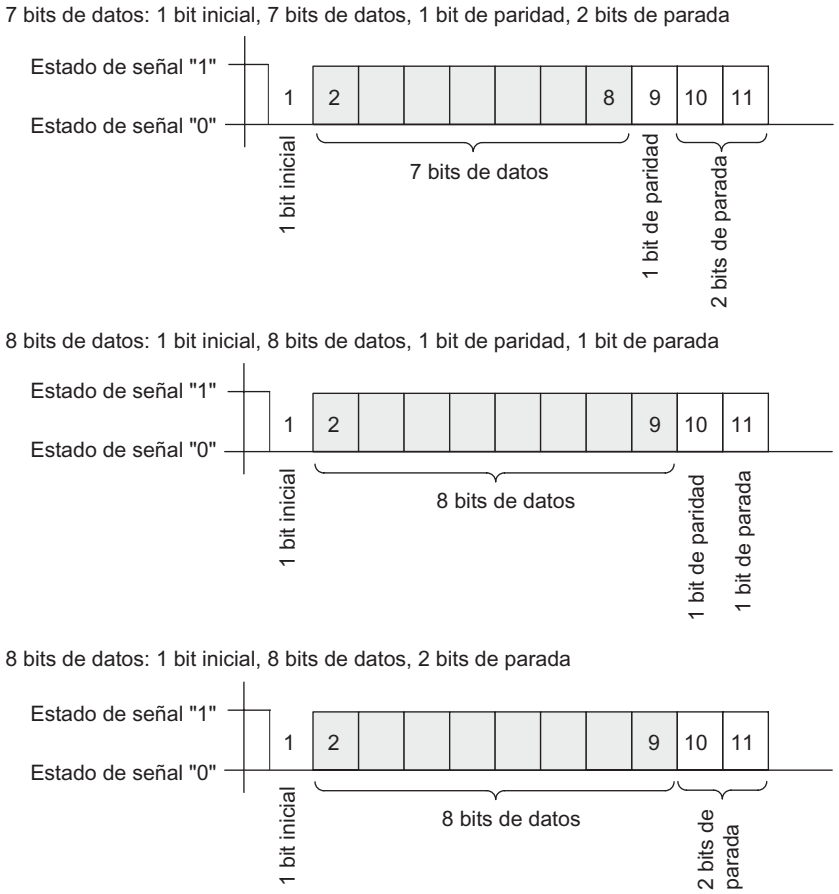


Figura 2-7 Trama de caracteres de 11 bits

Tiempo de retardo de caracteres

La figura siguiente muestra el tiempo máximo que puede transcurrir entre dos caracteres recibidos dentro de un telegrama. Se denomina tiempo de retardo entre caracteres.

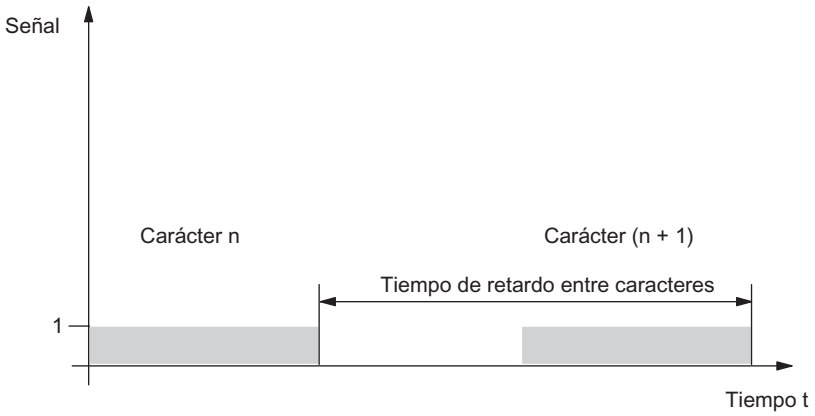


Figura 2-8 Tiempo de retardo de caracteres

2.6.3 Procedimiento de transmisión en el acoplamiento punto a punto

En el proceso de transmisión de datos es necesario que todos los interlocutores se atengan a determinadas normas para el desarrollo y la ejecución del intercambio de datos. ISO ha fijado un modelo de 7 capas que está reconocido como la base de una normativa mundial de protocolos de transmisión.

Protocolo

Todos los interlocutores implicados en el proceso de transmisión de datos deben atenerse a determinadas normas para el desarrollo y la ejecución del intercambio de datos. Dichas normas se conocen bajo el nombre de protocolos.

Cada protocolo determina:

- **El modo de operación**

  Modo semidúplex o dúplex

- **La iniciativa**

  Convenciones que determinan qué interlocutor y bajo qué condiciones puede iniciar la transmisión de datos.

- **Los caracteres de control**

  Definición de los caracteres de control utilizados en la transmisión de datos

- **La trama de caracteres**

  Determina qué trama de caracteres se va a utilizar para la transmisión de datos.

- **La protección de los datos**

  Definición del procedimiento de protección de datos

- **El tiempo de retardo entre caracteres**

  Determina el tiempo en el que debe llegar un carácter de recepción.

- **La velocidad de transmisión**

  Definición de la velocidad de transferencia en bits/s

Procedimiento

El proceso que se desarrolla durante una transmisión de datos se denomina procedimiento.

Modelo de referencia ISO de 7 capas

El modelo de referencia define el comportamiento externo de los interlocutores. Cada capa del protocolo está comprendida en la siguiente capa inferior, a excepción de la última capa inferior.

Cada una de las capas están definidas de la siguiente manera:

1. **Capa física**
   - Requisitos físicos para la transmisión de datos, como p. ej. medio de transmisión, velocidad de transmisión
2. **Capa de enlace de datos**
   - Procedimiento para proteger la transmisión de datos
   - Procedimiento de acceso
3. **Capa de red**
   - Definición de las vías de comunicación
   - y direccionamiento para la transmisión de datos entre dos interlocutores
4. **Capa de transporte**
   - Procedimiento de detección de fallos
   - Medidas de corrección
   - Procedimiento handshake
5. **Capa de sesión**
   - Establecimiento de la transmisión de datos
   - Ejecución
   - Fin de la transmisión de datos
6. **Capa de presentación**
   - Conversión del modo de representación normalizado del sistema de comunicación a una forma específica para el equipo (normas de interpretación de los datos)
7. **Capa de aplicación**
   - Definición de las tareas de comunicación y de las funciones necesarias para su ejecución

Procesamiento de los protocolos

El interlocutor emisor recorre los protocolos desde la capa superior (nº 7, orientada a la aplicación) hasta la inferior (nº 1 especificaciones físicas), mientras que el interlocutor receptor procesa los protocolos en orden ascendente desde la capa 1.

No todos los protocolos deben considerar las 7 capas. Si ambos interlocutores, emisor y receptor, hablan el mismo lenguaje, sobra la capa 6.

2.6.4 Seguridad de transmisión

Principio

La seguridad de transmisión es muy importante en el proceso de transferencia de datos y para la selección del procedimiento de transmisión. Por norma general puede decirse que cuantas más capas del modelo de referencia se ejecuten, mayor será la seguridad de transmisión.

Protocolos compatibles

La figura siguiente muestra cómo deben integrarse en el modelo de referencia ISO los protocolos ASCII y 3964(R) soportados por el módulo interfaz ET 200S 1SI.

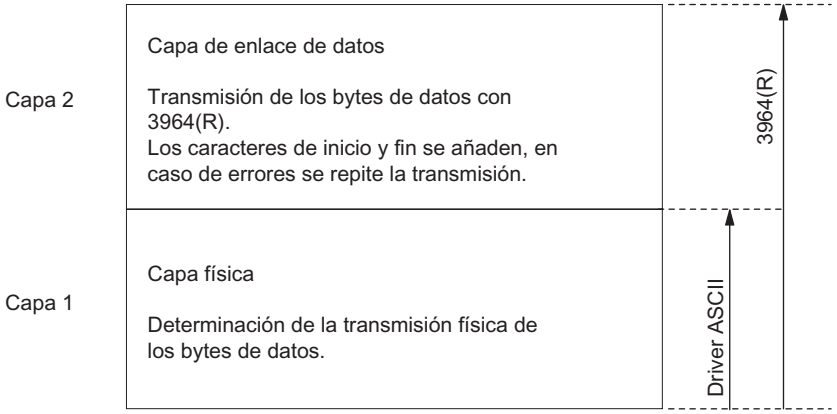

Capa 2
Capa de enlace de datos
Transmisión de los bytes de datos con 3964(R).
Los caracteres de inicio y fin se añaden, en caso de errores se repite la transmisión.
Capa 1
Capa física
Determinación de la transmisión física de los bytes de datos.
3964(R)
Driver ASCII

Figura 2-9 Integración de los protocolos soportados en el modelo de referencia

Seguridad de transmisión con el driver ASCII

Siga las pautas siguientes para aumentar la protección de los datos cuando utilice el driver ASCII:

- En un transporte de datos con el driver ASCII, aparte del uso del bit de paridad (que también puede desmarcarse en función de la parametrización de la trama de caracteres), no hay ninguna otra medida de protección de los datos. Por ello, si bien las transferencias de datos con el driver ASCII son muy eficientes en lo que se refiere al flujo de datos, el transporte de datos no está asegurado.
- Mediante el uso del bit de paridad se garantiza la inversión de un bit en un carácter transmitido. Si se invierten dos o más bits de un carácter, el fallo ya no podrá detectarse.
- Si fuera necesario aumentar la seguridad de la transmisión, puede hacerse introduciendo una suma de verificación (checksum) y la indicación de longitud de un telegrama. Estas medidas debe tomarlas el usuario.
- Para aumentar la protección de los datos pueden introducirse telegramas de acuse en los telegramas de emisión y recepción. Esto también ocurre en los protocolos de comunicación de datos de alta calidad (véase el modelo de referencia ISO de 7 capas).

Seguridad de transmisión con 3964(R)

El procedimiento 3964(R) ofrece una mayor protección de los datos:

- La distancia Hamming con 3964(R) es de 3. La distancia Hamming es una medida para la seguridad de una transmisión de datos.
- Mediante el procedimiento 3964(R) se garantiza una elevada seguridad de transmisión en la línea de transmisión. La elevada seguridad de transmisión se consigue gracias a una composición y descomposición prefijadas del telegrama, así como a la inserción del carácter de comprobación de bloques (BCC).

Dependiendo de si se desea transmitir los datos con o sin carácter de comprobación de bloque, se distingue entre

- Transmisión de datos sin carácter de comprobación de bloque: **3964**
- Transmisión de datos con carácter de comprobación de bloque: **3964R**

En las descripciones e indicaciones que se refieren a ambos métodos de transmisión de datos se utiliza en este manual la denominación **3964 (R)**.

Límites de capacidad con 3964(R)

- El procesamiento posterior de los datos de emisión/recepción en el programa en el interlocutor no está garantizado. Sólo puede garantizarse con un mecanismo de confirmación que debe programarse aparte.
- ¡La comprobación de bloques del procedimiento 3964R (enlace EXOR) no permite reconocer la falta de ceros (como carácter completo), ya que en el enlace EXOR un cero no afecta en modo alguno al cálculo final!

  La pérdida de un carácter completo (que en ese caso debe ser exactamente un cero), es muy improbable, pero puede ocurrir p. ej., en caso de que las condiciones de transmisión sean muy desfavorables.

  Para impedir que se produzcan este tipo de fallos, proteja la comunicación transmitiendo, además de los datos, la longitud del telegrama para que ésta sea evaluada por el interlocutor.

2.7 Transmisión de datos con el procedimiento 3964(R)

2.7.1 Conceptos básicos de la transmisión de datos con el procedimiento 3964(R)

Principio

El procedimiento 3964(R) controla la transmisión de datos en un acoplamiento punto a punto entre el módulo ET 200S y un interlocutor. Además de la capa física (capa 1), el procedimiento 3964(R) contiene la capa de enlace de datos (capa 2).

Caracteres de control

En la transmisión de datos, el procedimiento 3964(R) añade a los datos útiles caracteres de control (capa de enlace de datos). Estos caracteres de control permiten que el interlocutor controle si los datos le han llegado completos y sin fallos.

En el procedimiento 3964(R) se evalúan los siguientes caracteres de control:

- **STX**: Start of Text;

  Inicio de la secuencia de caracteres que debe transmitirse
- **DLE**: Data Link Escape;

  Conmutación de transferencia de datos
- **ETX**: End of Text;

  Fin de la secuencia de caracteres que debe transmitirse
- **BCC**: Block Check Character (sólo con 3964R);

  Carácter de comprobación de bloque
- **NAK**: Negative Acknowledge;

  Confirmación negativa

Nota

Si se transmite el carácter DLE como carácter de información, éste se enviará doblemente para distinguir el carácter de control DLE durante el establecimiento y la desconexión de la comunicación en la línea de transferencia (duplicado de DLE). El receptor anula el duplicado del carácter DLE.

Prioridad

En el procedimiento 3964(R) un interlocutor debe tener asignada una prioridad mayor que el otro. Si ambos interlocutores establecen simultáneamente la comunicación, aquel que tenga menor prioridad retirará su petición de envío.

Suma de comprobación de bloque

Con el protocolo de transmisión 3964R, la protección de los datos se incrementa mediante un carácter de comprobación de bloque (BCC = Block Check Character) enviado adicionalmente (véase la siguiente figura).

Telegrama:

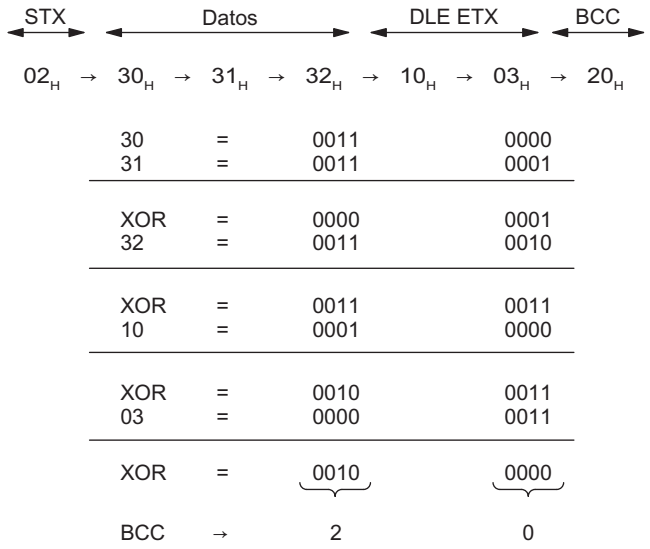


Figura 2-10 Suma de comprobación de bloque

La suma de comprobación de bloque es la paridad longitudinal par (operación lógica EXOR de todos los bytes de datos) de un bloque enviado o recibido. La formación empieza con el primer byte de datos útiles (1er byte del telegrama) tras el establecimiento de la conexión y finaliza después del carácter DLE ETX al interrumpir la conexión.

Nota

En un duplicado DLE, el carácter DLE se incluye dos veces en la formación del BCC.

2.7.2 Enviar datos con el procedimiento 3964(R)

Enviar datos con 3964(R)

En la figura siguiente se representa el desarrollo de la transmisión de datos al enviar con el procedimiento 3964(R).

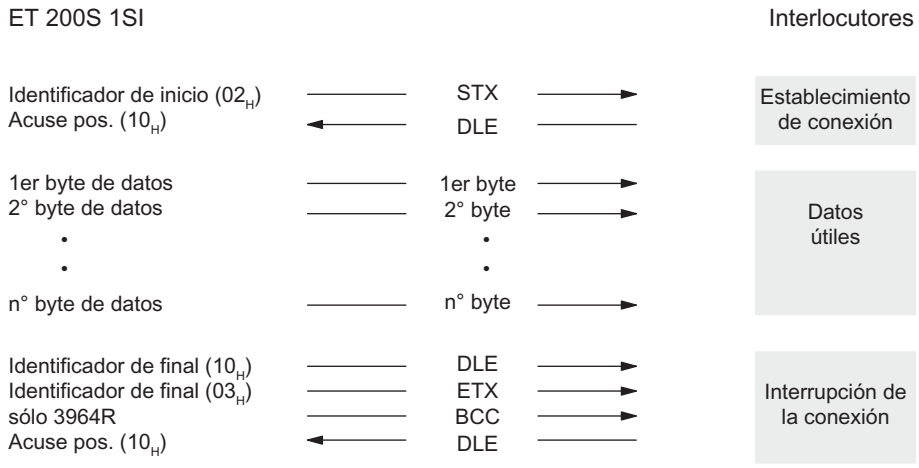


Figura 2-11 Intercambio de datos al transmitir con el procedimiento 3964(R)

Establecimiento de la conexión en el envío

Para establecer la conexión, el procedimiento 3964(R) envía el carácter de control STX. Si el interlocutor responde con el carácter DLE antes de que transcurra el tiempo de retardo del acuse (TRA), el procedimiento entra en la fase de envío.

Si, por el contrario, el interlocutor responde con NAK u otro carácter cualquiera (con la excepción de DLE), o si el tiempo de retardo del acuse transcurre sin que se produzca ninguna reacción, el procedimiento repite el establecimiento de la conexión. Tras el número parametrizado de intentos fallidos, el procedimiento interrumpe el establecimiento y envía el carácter NAK al interlocutor. El programa del sistema registra el error en el bloque de función S_SEND (parámetro de salida STATUS).

Enviar datos

Si la conexión se establece correctamente, se envían al interlocutor los datos útiles incluidos en el búfer de salida del módulo ET 200S con los parámetros de transmisión seleccionados. El interlocutor controla la distancia temporal de los caracteres entrantes. La distancia entre dos caracteres no debe ser mayor al tiempo de retardo entre caracteres (TRC).

Interrupción de la conexión en el envío

Si el interlocutor envía el carácter NAK durante un envío, el procedimiento interrumpe el bloque y lo repite como se ha descrito anteriormente. Si el interlocutor envía cualquier otro carácter, el procedimiento espera primero a que transcurra el tiempo de retardo de caracteres y a continuación envía el carácter NAK para que el interlocutor pase a estado de reposo. Entonces el procedimiento comienza de nuevo la fase de envío con el establecimiento de la conexión STX.

Tras el envío del contenido del búfer, el procedimiento añade los caracteres DLE, ETX y, **sólo en el caso de 3964(R)**, la suma de verificación de bloques BCC como carácter de fin, y espera un carácter de acuse. Si el interlocutor envía el carácter DLE dentro del tiempo de retardo de acuse, el bloque de datos se recibe sin errores. Si, por el contrario, el interlocutor responde con NAK, otro carácter cualquiera (con la excepción de DLE), un carácter erróneo, o si el tiempo de retardo de acuse transcurre sin que se produzca una reacción, el procedimiento comienza de nuevo el envío con el establecimiento de la conexión STX.

Tras un número preestablecido de intentos de envío del bloque de datos, el procedimiento interrumpe el proceso y envía el carácter NAK al interlocutor. El programa del sistema registra el error en el bloque de función S_SEND (parámetro de salida STATUS).

2.7.3 Recibir datos con el procedimiento 3964(R)

Recibir datos con 3964(R)

En la figura siguiente se representa el desarrollo de la transmisión de datos al recibir con el procedimiento 3964(R).

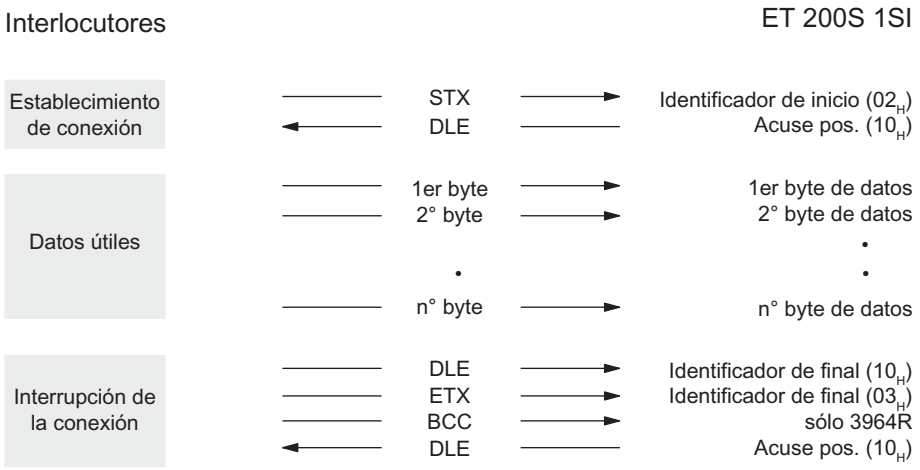


Figura 2-12 Intercambio de datos en la recepción con el procedimiento 3964(R)

Establecimiento de la conexión en la recepción

En estado de reposo, cuando no debe procesarse ninguna orden de envío, el procedimiento espera a que el interlocutor establezca la conexión.

Si el procedimiento recibe cualquier carácter (con la excepción de STX o NAK) durante el modo de reposo, espera primero a que transcurra el tiempo de retardo entre caracteres (TRC) y a continuación envía el carácter NAK.

Recibir datos

Si el procedimiento recibe el carácter STX y dispone de un búfer de recepción vacío, responde con DLE. Los caracteres de recepción entrantes se depositan en el búfer de recepción. Si se reciben dos caracteres DLE consecutivos, sólo uno de ellos se registra en el búfer.

Después de cada carácter recibido, se espera al siguiente durante el tiempo de retardo de caracteres. Si éste transcurre sin ninguna recepción, se envía el carácter NAK al interlocutor. El programa del sistema registra entonces el error en el bloque de función S_RCV (parámetro de salida STATUS).

Si durante el establecimiento de la conexión con STX no hay ningún búfer de recepción disponible, se inicia un tiempo de espera de 400 ms. Si una vez transcurrido este tiempo continúa sin haber ningún búfer de recepción disponible, el programa del sistema registra el error (mensaje de error en la salida STATUS del FB). El procedimiento envía un carácter NAK y regresa al estado de reposo. En caso contrario, el procedimiento envía el carácter DLE y recibe los datos como se ha descrito.

Interrupción de la conexión en la recepción

Si durante la recepción se producen errores de transmisión (carácter perdido, error de trama, error de paridad, etc.), se continúa recibiendo hasta que se produce la interrupción de la conexión y a continuación se envía el carácter NAK al interlocutor. A continuación se espera una repetición. Si no es posible recibir el bloque sin errores tras el número de intentos de transmisión especificado en la parametrización, o si el interlocutor no inicia la repetición dentro de un tiempo de espera de 4 s, el procedimiento interrumpe la recepción. El programa del sistema registra entonces el error en el bloque de función S_RCV (parámetro de salida STATUS).

Cuando el procedimiento **3964(R)** detecta la secuencia de caracteres DLE ETX, finaliza la recepción y envía el carácter DLE al interlocutor tras recibir un bloque con éxito. Si, por el contrario, se produce un error de transmisión, envía el carácter NAK al interlocutor. A continuación se espera una repetición.

Cuando el procedimiento **3964(R)** detecta la secuencia de caracteres DLE ETX BCC, finaliza la recepción. Compara el carácter de comprobación de bloques BCC con la paridad longitudinal calculada internamente. Si el carácter de comprobación de bloques es correcto y no ha ocurrido ningún otro error en la recepción, el procedimiento 3964(R) envía el carácter DLE y vuelve al estado de reposo. Si hay un error en el BCC o si se produce cualquier otro error de recepción, se envía el carácter NAK al interlocutor. A continuación se espera una repetición.

Nota

En cuanto está preparado, el procedimiento 3964(R) comienza a enviar el carácter NAK al interlocutor para que éste pase a estado de reposo.

Parámetros del procedimiento

Para ambos interlocutores de una comunicación 3964(R) deben seleccionarse ajustes idénticos para los siguientes parámetros del procedimiento:

- Tiempo de retardo de caracteres
- Tiempo de retardo de confirmación
- Intentos de establecimiento
- Intentos de transmisión

Excepción:

Si utiliza el módulo ET 200S 1SI con prioridad baja, parametrice el número de intentos de establecimiento para ese módulo como mínimo "1" más bajo que en el interlocutor para solucionar con mayor rapidez los posibles conflictos de inicialización (véase el apartado Conflicto de inicialización del capítulo Tratamiento de errores con el procedimiento 3964(R) (Página 41)).

2.7.4 Tratamiento de errores con el procedimiento 3964(R)

Tratamiento de datos erróneos

La figura siguiente muestra la estructura del proceso de tratamiento de los datos erróneos con el procedimiento 3964(R).

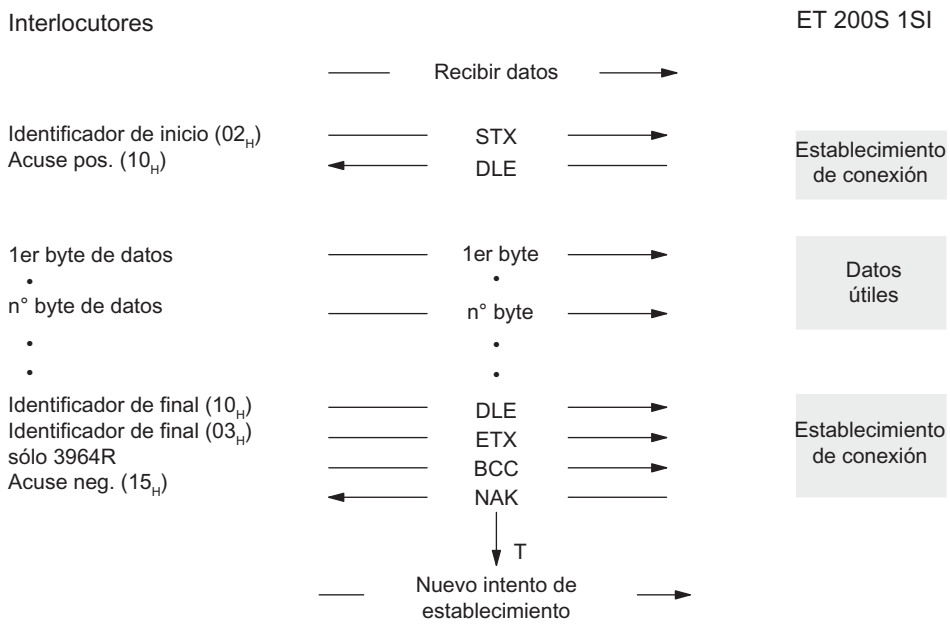


Figura 2-13 Intercambio de datos en la recepción de datos erróneos

Tras la recepción de DLE, ETC y BCC, el módulo ET 200S 1SI compara el BCC del interlocutor con el valor propio creado internamente. Si el BCC es correcto y no ha surgido ningún otro error de recepción, el módulo ET 200S 1SI responde con DLE.

De lo contrario, el módulo responde con NAK y espera el tiempo de espera de bloque (T) de 4 s a un nuevo intento. Si el bloque no puede recibirse una vez transcurrido el número parametrizado de intentos de transmisión o no se realiza ningún nuevo intento durante el tiempo de espera de bloque, el módulo ET 200S 1SI cancela la recepción.

Conflicto de inicialización

La figura siguiente muestra el proceso de transmisión de datos en caso de conflicto de inicialización.

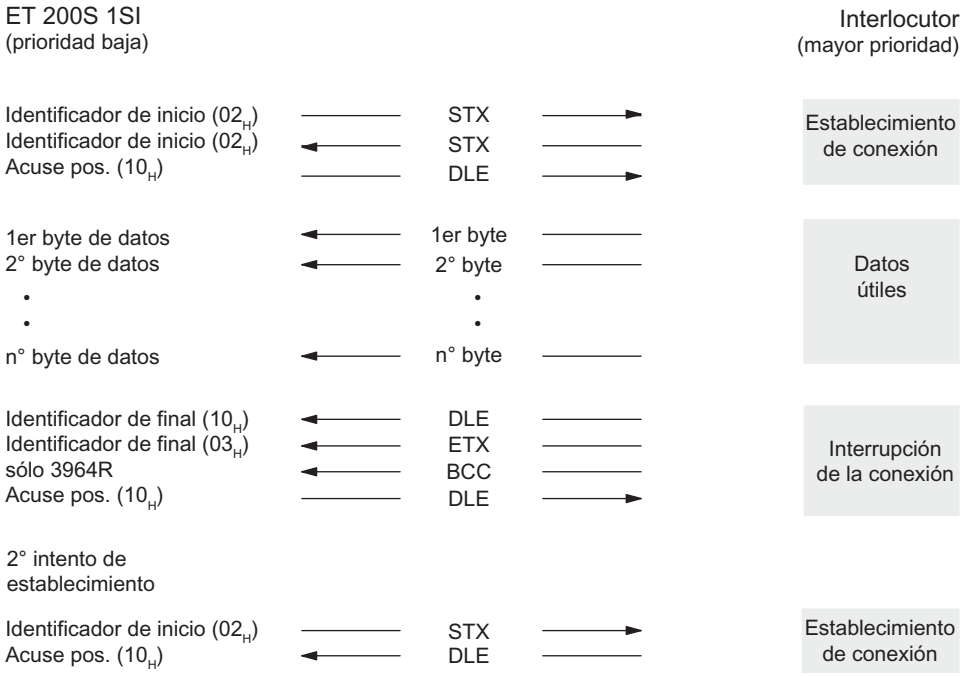


Figura 2-14 Intercambio de datos en caso de conflicto de inicialización

Si un equipo no responde a la petición de envío (carácter STX) del interlocutor dentro del tiempo de retardo de acuse (TRA) con la confirmación DLE o NAK, pero sí con el carácter STX, se produce un conflicto de inicialización. Ambos equipos desean ejecutar una petición de envío existente. El equipo que tiene menor prioridad retira su petición y responde con el carácter DLE. El equipo con mayor prioridad envía sus datos de la forma descrita anteriormente. Una vez interrumpida la conexión, el equipo con menor prioridad puede ejecutar su petición de envío.

Para solucionar el conflicto de inicialización, hay que parametrizar los interlocutores con diferentes prioridades.

Fallo de procedimiento

El procedimiento detecta tanto los fallos producidos por un comportamiento erróneo del interlocutor, como los debidos a fallos en la línea.

En ambos casos, primero se intenta conseguir un buen envío/una buena recepción del bloque de datos repitiendo el proceso. Si el bloque de datos no puede enviarse o recibirse sin errores hasta el número máximo de repeticiones (o si se produce un estado de error nuevo), el procedimiento cancela el envío o la recepción. Registra el número de error para el primer error reconocido y pasa al estado de reposo. Estos mensajes de error se visualizan en la salida STATUS del FB.

Si en la salida STATUS del FB hay con frecuencia un número de error para repeticiones de envío y recepción, esto sugiere que hay perturbaciones ocasionales en el tráfico de datos. Sin embargo, la cantidad de repeticiones lo compensa. En este caso le recomendamos que compruebe posibles fuentes que puedan estar afectando a la transmisión, ya que con las múltiples repeticiones se reduce tanto la tasa de datos útiles como la seguridad. La causa del fallo también puede ser un comportamiento erróneo del interlocutor.

En caso de BREAK en la línea de recepción (línea de recepción interrumpida), se comunica un estado BREAK (indicador BREAK mediante una alarma de diagnóstico del módulo ET 200S (véase el apartado Diagnóstico (Página 110)). No se inicia ninguna repetición. El estado BREAK se anula automáticamente en cuanto se restablece la conexión en la línea.

Para todos los errores de transmisión detectados (carácter perdido, fallo de trama o de paridad) se notifica un número estándar, independientemente de que el fallo se haya producido durante la transmisión o recepción de un bloque de datos. No obstante, el fallo sólo se notifica si los reintentos tampoco han dado resultado.

2.8 Transmisión de datos con el driver ASCII

2.8.1 Conceptos básicos de la transmisión de datos con el driver ASCII

Introducción

El driver ASCII controla la transmisión de datos en un acoplamiento punto a punto entre el módulo ET 200S 1SI y un interlocutor. El driver ASCII contiene la capa física (capa 1).

La configuración de los telegramas se mantiene abierta mediante la transmisión de todo el telegrama de envío al módulo ET 200S 1SI por parte del usuario de S7. Para la recepción, debe parametrizarse el criterio de fin de un telegrama. La configuración de los telegramas de envío puede ser diferente de la configuración de los telegramas de recepción.

Con el driver ASCII pueden enviarse y recibirse datos con cualquier configuración (todos los caracteres ASCII imprimibles así como el resto de caracteres desde 00 hasta FF_H (en tramas de caracteres con 8 bits de datos) y desde 00 hasta $7F_H$ (en tramas de caracteres con 7 bits de datos)).

Consulte también

Principios de la comunicación a través de bloques de función (Página 67)

Propiedades de arranque y estados operativos (Página 89)

2.8.2 Enviar datos con el driver ASCII

Enviar datos con el driver ASCII

Para enviar, al llamar el bloque de función S_SEND se debe indicar como parámetro "LEN" el número de bytes de datos útiles que deben enviarse. Los datos útiles deben contener los caracteres de inicio y fin que pudieran ser necesarios.

Si trabaja con el criterio de fin "Tiempo de retardo de caracteres transcurrido", el driver ASCII realiza una pausa entre dos telegramas también al enviar. Puede acceder en todo momento al FB S_SEND, pero el driver ASCII no inicia la salida hasta que desde el último telegrama enviado haya transcurrido un tiempo superior al tiempo de retardo entre caracteres parametrizado.

Nota

Al parametrizar los caracteres XON/XOFF para el control de flujo, los datos útiles no deben contener ninguno de los caracteres parametrizados XON o XOFF. Los ajustes predeterminados son DC1 = 11_H para XON y DC3 = 13_H para XOFF.

Enviar datos

La figura siguiente muestra los procesos que se desarrollan al enviar.

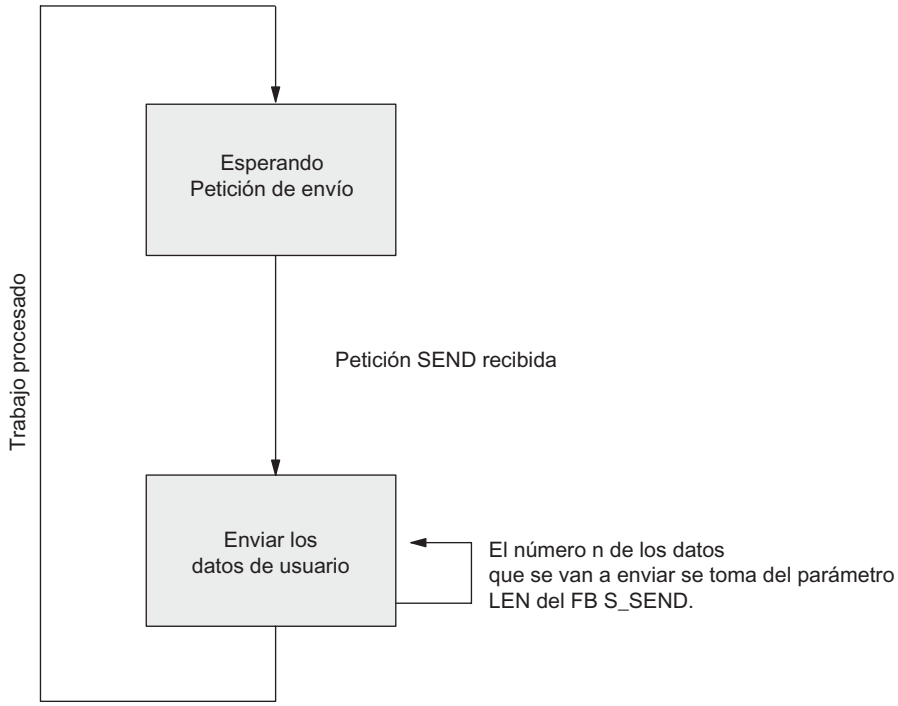


Figura 2-15 Desarrollo del proceso de envío

2.8.3 Recibir datos con el driver ASCII

Recibir datos con el driver ASCII

Al transmitir datos con el driver ASCII puede elegir entre tres criterios de fin distintos. El criterio de fin determina el momento en que un telegrama ha sido recibido de forma completa. Los criterios de fin posibles son:

- **Transcurrido el tiempo de retardo entre caracteres**

  El telegrama no tiene una longitud específica ni caracteres de fin definidos, el final del telegrama se determina mediante una pausa en la línea (tiempo de retardo entre caracteres transcurrido). Abajo encontrará los valores mínimos de las diferentes velocidades de transmisión.

- **Recepción del/de los carácter(es) de fin**

  Al final del telegrama hay uno o dos caracteres de fin definidos.

- **Recepción de un número determinado de caracteres**

  La longitud de los telegramas de recepción es siempre la misma.

Transparencia de códigos

La transparencia de códigos del procedimiento depende de la elección del criterio de fin parametrizado y del control de flujo:

- Con uno o dos caracteres de fin
  - No hay transparencia de códigos
- Criterio de fin Tiempo de retardo entre caracteres o Longitud fija de telegrama
  - Con transparencia de código
- Si se utiliza el control de flujo XON/XOFF, no hay transparencia de código.

El término "transparencia de código" significa que en los datos útiles pueden darse todas las combinaciones de caracteres posibles sin que se detecte el criterio de fin.

Tiempo mínimo de retardo entre caracteres según la velocidad de transmisión

El valor mínimo para el tiempo de retardo entre caracteres depende de la velocidad de transmisión. La tabla siguiente lista el tiempo mínimo de retardo entre caracteres en ms para las diferentes velocidades de transmisión.

Tabla 2- 7 Tiempo mínimo de retardo entre caracteres

Velocidad de transmisión	Tiempo mínimo de retardo entre caracteres
115	365 ms
300	130 ms
600	65 ms
1.200	32 ms
2.400	16 ms
4.800	8 ms
9.600	4 ms
19.200	2 ms
38.400	1 ms
57.600	1 ms
76.800	1 ms
115.200	1 ms

Búfer de recepción del módulo ET 200S

El búfer de recepción del módulo interfaz ET 200S 1SI abarca 4096 bytes. En la parametrización se puede especificar si debe borrarse el búfer de recepción durante el arranque, y si debe evitarse que los datos del búfer de recepción se sobrescriban. Asimismo, se puede activar o bloquear el respaldo de los telegramas recibidos.

El búfer de recepción del módulo interfaz serie ET 200S 1SI es un búfer circular:

- Si se registran varios telegramas en el búfer de recepción del módulo ET 200S 1SI, se aplica la siguiente norma: Siempre se transmite el telegrama más antiguo del módulo ET 200S 1SI a la CPU.
- Para transmitir siempre únicamente el telegrama más reciente a la CPU, bloquee Telegramas Dinámicos **y** desconecte la protección de sobrescritura.

Nota

Si la lectura continua de los datos recibidos en el programa de usuario se interrumpe durante un tiempo, al volver a solicitar los datos recibidos puede ocurrir que la CPU reciba primero telegramas antiguos del módulo 200S 1SI y después el último telegrama.

El telegrama antiguo es el telegrama que, cuando se produjo la interrupción, estaba de camino entre el ET 200S 1SI y la CPU o bien que ya había sido recibido por el FB.

2.8.4 Criterios de fin para la transmisión de datos con el driver ASCII

Criterio de fin "Transcurrido el tiempo de retardo entre caracteres"

En la recepción de datos se detecta el final del telegrama una vez que ha transcurrido el tiempo de retardo entre caracteres. Los datos recibidos son aceptados por la CPU con el bloque de función S_RCV.

El tiempo de retardo entre caracteres debe ajustarse de tal modo que transcurra con seguridad entre dos telegramas consecutivos. Asimismo, ha de ser lo suficientemente largo como para garantizar que en las pausas de envío del interlocutor dentro de un mismo telegrama no se reconozca por error el final del mismo.

La siguiente figura muestra el desarrollo del proceso de recepción con criterio de fin "Transcurrido el tiempo de retardo entre caracteres".

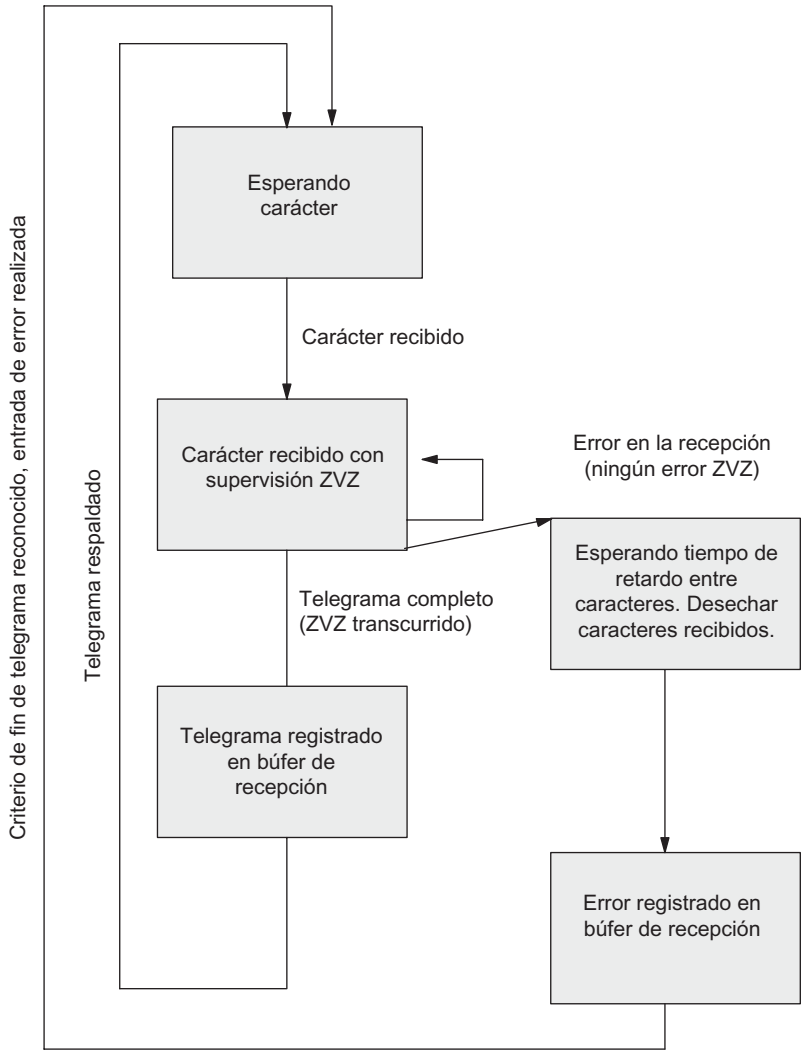

Esperando carácter
Carácter recibido
Carácter recibido con supervisión ZVZ
Error en la recepción (ningún error ZVZ)
Esperando tiempo de retardo entre caracteres. Desechar caracteres recibidos.
Telegrama completo (ZVZ transcurrido)
Telegrama registrado en búfer de recepción
Error registrado en búfer de recepción
Telegrama respaldado
Criterio de fin de telegrama reconocido, entrada de error realizada

Figura 2-16 Desarrollo del proceso de recepción con el criterio de fin "Transcurrido el tiempo de retardo entre caracteres"

Criterio de fin "Carácter de fin"

En la recepción de datos se detecta el final del telegrama en el momento en que se recibe(n) el/los carácter(es) de fin parametrizado(s). Los datos recibidos son aceptados por la CPU, inclusive el carácter de fin, con el bloque de función S_RCV.

El transcurso del tiempo de retardo entre caracteres durante la recepción implica el final de la recepción. Se genera un mensaje de error y el fragmento de telegrama se rechaza.

Cuando se trabaja con caracteres de fin, la transferencia no tiene transparencia de códigos y debe excluirse la posibilidad de que el/los identificador(es) de fin aparezcan en los datos útiles.

La siguiente figura muestra el desarrollo del proceso de recepción con criterio de fin "Carácter de fin".

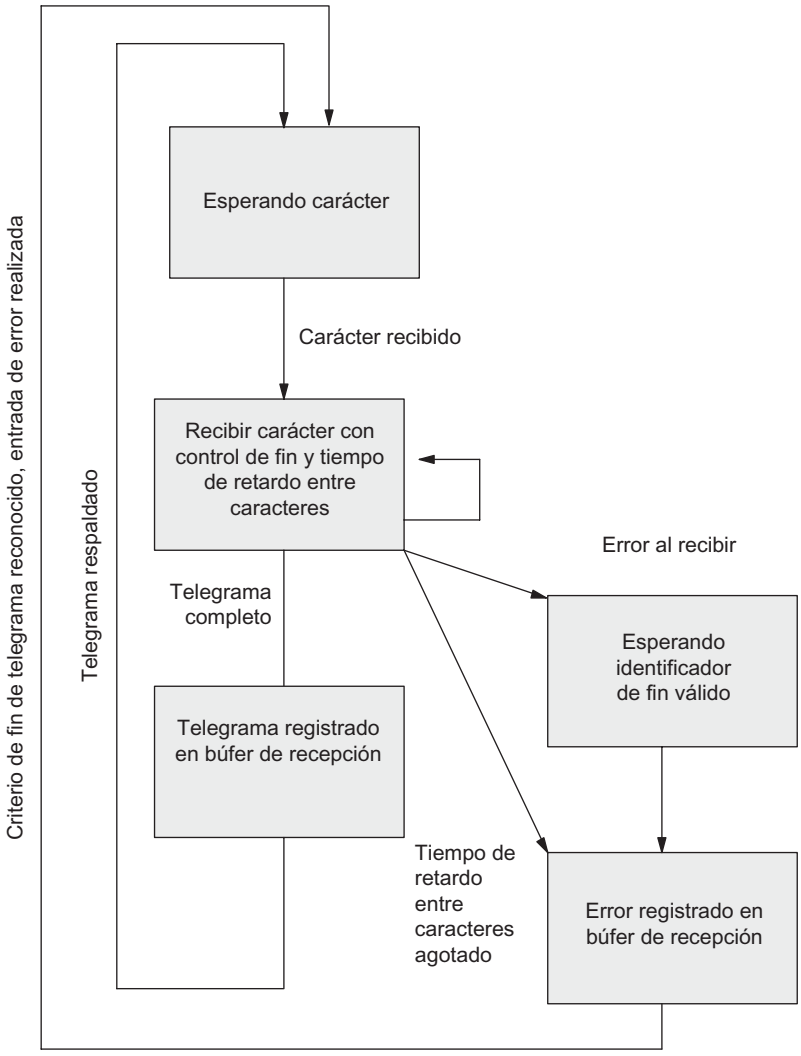

Esperando carácter
Carácter recibido
Recibir carácter con control de fin y tiempo de retardo entre caracteres
Error al recibir
Telegrama completo
Esperando identificador de fin válido
Telegrama registrado en búfer de recepción
Tiempo de retardo entre caracteres agotado
Error registrado en búfer de recepción
Telegrama respaldado
Criterio de fin de telegrama reconocido, entrada de error realizada

Figura 2-17 Proceso de recepción con el criterio de fin "Carácter de fin"

Criterio de fin "Longitud de telegrama predeterminada"

En la recepción de datos se detecta el final del telegrama en el momento en que se ha recibido el número de caracteres parametrizado. Los datos recibidos son aceptados por la CPU con el bloque de función S_RCV.

Si el tiempo de retardo entre caracteres termina antes de que se alcance el número de caracteres parametrizado, se finaliza la recepción. Se genera un mensaje de error y el fragmento de telegrama se rechaza.

La siguiente figura muestra el desarrollo del proceso de recepción con criterio de fin "Longitud de telegrama predeterminada".

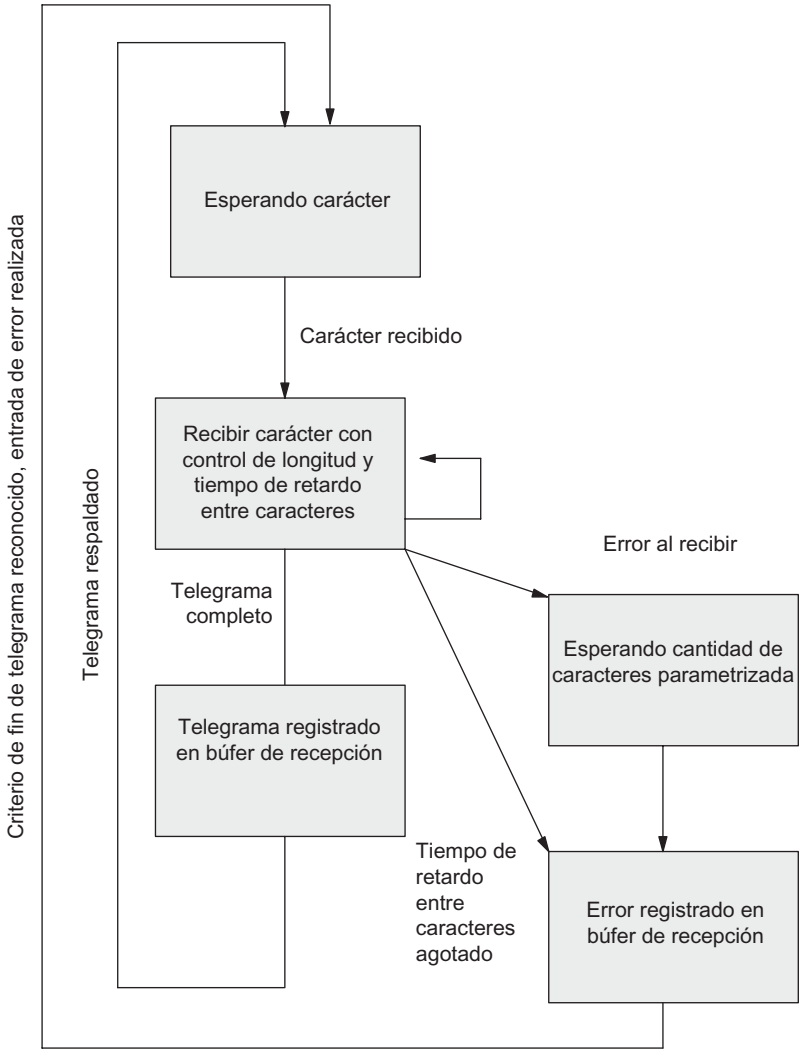

Esperando carácter
Carácter recibido
Recibir carácter con control de longitud y tiempo de retardo entre caracteres
Error al recibir
Telegrama completo
Esperando cantidad de caracteres parametrizada
Telegrama registrado en búfer de recepción
Tiempo de retardo entre caracteres agotado
Error registrado en búfer de recepción
Telegrama respaldado
Criterio de fin de telegrama reconocido, entrada de error realizada

Figura 2-18 Esquema de desarrollo en la recepción con criterio de fin "longitud de telegrama predeterminada"

2.8.5 Señales cualificadoras RS-232C para la transmisión de datos con el driver ASCII

Señales cualificadoras RS 232C

El módulo ET 200S 1SI soporta las siguientes señales cualificadoras RS232C:

- DCD (Entrada) Data Carrier detect; soporte de datos detectado
- DTR (Salida) Data terminal ready; ET 200S 1SI listo para funcionar
- DSR (Entrada) Data set ready; interlocutor listo para funcionar
- RTS (Salida) Request to send; ET 200S 1SI listo para enviar
- CTS (Entrada) Clear to send; el interlocutor puede recibir datos del módulo ET 200S 1SI (respuesta a RTS = ON del ET 200S 1SI)

Después de conectar el módulo ET 200S 1SI, las señales de salida están en estado OFF (inactivas).

El funcionamiento de las señales de control DTR/DSR y RTS/CTS puede parametrizarse con la interfaz de parametrización o forzarse con las funciones (FCs) del programa de usuario.

Manejo de las señales cualificadoras RS 232C

Las señales cualificadoras RS 232C pueden utilizarse de la siguiente manera:

- Con manejo automático parametrizado para todas las señales cualificadoras RS 232C
- Con flujo de datos parametrizado (RTS/CTS)
- A través de los bloques de función (FBs) S_VSTAT y S_VSET

Nota

Al parametrizar un funcionamiento automático de las señales cualificadoras RS 232C, no es posible ni un control del flujo de datos con RTS/CTS ni un forzado de RTS y DTR mediante el FB S_VSET.

Al parametrizar un control del flujo de datos con RTS/CTS, no es posible un forzado de RTS mediante el FB S_VSET.

En cambio, sí es posible leer todas las señales cualificadoras RS 232C mediante el FB S_VSTAT.

A continuación se explica el procedimiento básico para controlar y evaluar las señales cualificadoras de RS 232C.

Manejo automático de las señales cualificadoras RS232C

El manejo automático de las señales cualificadoras RS232C en el módulo ET 200S 1SI está implementado del modo siguiente:

- En cuanto el módulo ET 200S 1SI se ha pasado a un modo de manejo automático de las señales cualificadoras RS232C mediante la parametrización, ajusta las líneas RTS a OFF y DTR a ON (ET 200S 1SI listo para funcionar).

  Sólo es posible enviar y recibir telegramas tras poner la línea DTR a ON. Mientras DTR permanece en OFF, no se pueden recibir datos vía la interfaz RS 232C. Cualquier petición de envío se interrumpe con el correspondiente mensaje de error.
- Si está pendiente una **petición de envío**, RTS pasa al estado ON y se inicia el tiempo de espera de salida de datos parametrizado. Una vez transcurrido el tiempo de salida de datos, y con CTS = ON, los datos se envían a través de la interfaz RS 232C.
- Si durante el envío la línea CTS no pasa a ON durante el tiempo de espera, o si durante el proceso de salida, CTS cambia al estado OFF, se interrumpe el envío mostrando el mensaje correspondiente.
- Tras la emisión de los datos y una vez transcurrido el tiempo de RTS a OFF parametrizado, la línea RTS pasa al estado de inactividad. El módulo ET 200S 1SI no espera que CTS pase a OFF.
- Es posible **recibir** datos vía la interfaz RS 232C, tan pronto como se active la línea DSR (ON) . Si el búfer de recepción del módulo ET 200S 1SI está a punto de desbordarse, el módulo ET 200S 1SI no reacciona.
- Al cambiar de DSR = ON a OFF se interrumpe tanto el envío en curso como la recepción de datos con un mensaje de error.

Nota

Al parametrizar un funcionamiento automático de las señales cualificadoras RS 232C, no es posible ni un control del flujo de datos con RTS/CTS ni un forzado de RTS y DTR mediante el FB S_VSET.

Cronograma

La siguiente figura muestra el desarrollo temporal de una petición de envío.

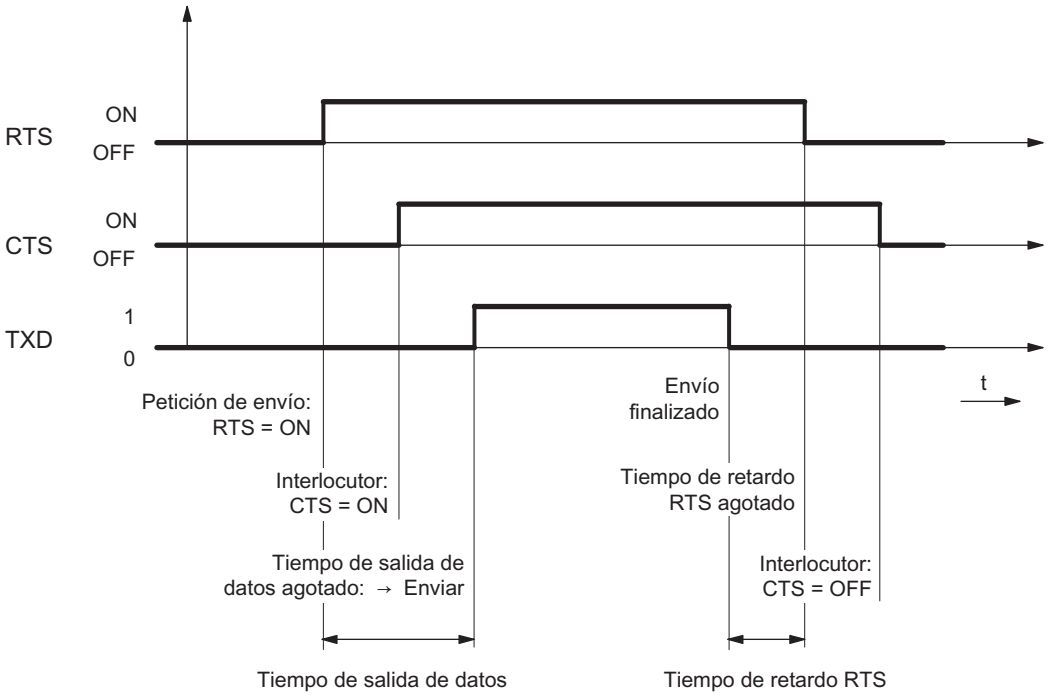


Figura 2-19 Cronograma para el manejo automático de las señales cualificadoras RS 232C

Control de flujo de datos/procedimiento handshake

Los procedimientos handshake controlan el flujo de datos entre dos interlocutores. El procedimiento handschake evita que se pierdan datos durante la transmisión con equipos que trabajan a diferente velocidad. Se distinguen dos procedimientos:

- Handshake de software (p. ej. XON/XOFF)
- Handshake de hardware (p. ej. RTS/CTS)

El control del flujo de datos del módulo ET 200S 1SI está implementado del modo siguiente:

- En cuanto el módulo ET 200S 1SI pasa a un modo de operación con control de flujo mediante la parametrización, envía el carácter XON o ajusta la línea RTS a ON.
- Al alcanzar el número de telegramas parametrizado o bien los 50 caracteres antes de que se desborde el búfer de recepción (capacidad del búfer de recepción: 4096 bytes), el módulo ET 200S 1SI envía el carácter XOFF o pone la línea RTS en estado OFF. Si a pesar de ello el interlocutor continúa enviando, se genera un mensaje de error si el búfer de recepción se desborda. Los datos recibidos del último telegrama se descartan.

- En cuanto la CPU S7 recoge un telegrama y el búfer de recepción está listo para recibir, el módulo ET 200S 1SI envía el carácter XON o ajusta la línea RTS a ON.
- Si el módulo ET 200S 1SI recibe el carácter XOFF o si la señal de control CTS se ajusta a OFF, el módulo ET 200S 1SI interrumpe el proceso de envío. Si pasado un tiempo parametrizable determinado no se recibe ningún XON o si CTS no se ajusta a ON, el proceso de envío se interrumpe y se genera un mensaje de error (0708_H) en la salida STATUS de los bloques de función.

Leer/forzar a través de FB S_VSTAT y FB S_VSET

Mediante el bloque de función S_VSTAT puede determinarse el estado de cada señal cualificadora RS232C. Con el bloque de función S_VSET es posible forzar las señales de salida DTR y RTS. En el apartado Principios de la comunicación a través de bloques de función (Página 67) encontrará información sobre cómo utilizar los bloques de función como interfaz entre la CPU y el módulo ET 200S 1SI.

2.9 Configuración y parametrización del módulo interfaz serie

2.9.1 Configuración del módulo interfaz serie

Principio

Si se comunica con un maestro S7 con el módulo interfaz ET 200S 1SI mediante una red PROFIBUS, trabaje en la configuración de hardware de STEP 7 para ajustar el módulo en la red PROFIBUS y ajustar los parámetros de comunicación del módulo.

Si selecciona el módulo ET 200S 1SI en el catálogo de hardware y lo inserta en el ET 200S básico en la configuración de la red, la referencia del módulo, el número del slot y las direcciones de entradas y salidas se incluirán automáticamente en la tabla de configuración. A continuación, se puede abrir el cuadro de diálogo de propiedades del módulo ET 200S 1SI y ajustar el tipo de comunicación y otros parámetros.

2.9.2 Parametrización del driver ASCII

Principio

La tabla siguiente lista los parámetros que pueden ajustarse para el driver ASCII del módulo interfaz serie.

Tabla 2- 8 Parámetros del driver ASCII

Parámetro	Descripción	Rango de valores	Valor por defecto
Alarma de diagnóstico	Indique si el módulo crea una alarma de diagnóstico cuando surge un error grave.	• No • Sí	No
Activar reconocimiento BREAK	Si se produce una ruptura de la línea o no se ha conectado ningún cable de interfaz, el módulo notifica el mensaje de error "Break".	• No • Sí	No
Tipo de interfaz	Indique la interfaz eléctrica que debe emplearse (véanse los apartados Interfaz RS-232C (Página 24) y Interfaz RS-422/485 (Página 25).)	• RS-232C • RS-422 (dúplex) • RS-485 (semidúplex)	RS-232C

Parámetro	Descripción	Rango de valores	Valor por defecto
Preajuste dúplex y semidúplex de la línea de recepción	Indique el preajuste de la línea de recepción en los modos de operación RS-422 y RS-485. No en el modo de operación RS232C. La configuración "Nivel invertido" sólo es requerida en caso de repuesto para asegurar la compatibilidad.	RS422: R(A) 5V / R(B) 0V (BREAK) R(A) 0V / R(B) 5V Nivel invertido RS485: Ninguno R(A) 0V / R(B) 5V	RS422: R(A) 5V / R(B) 0V (BREAK) RS 485: R(A) 0V / R(B) 5V
Control de flujo de datos (con parámetros predeterminados; cambiar valores predeterminados en el programa de usuario)	Se pueden enviar y recibir datos con control del flujo de datos. Mediante el control del flujo de datos, la transmisión de datos se sincroniza cuando un interlocutor trabaja más rápidamente que el otro. Seleccione el tipo de control del flujo de datos y ajuste los parámetros correspondientes (véase el apartado Conceptos básicos de la transmisión de datos con el driver ASCII (Página 44)). Nota: Con la interfaz RS 485 no es posible el control de flujo de datos. El control del flujo de datos con "RTS/CTS" y "Manejo automático de las señales V24" sólo es posible con la interfaz RS232C.	• Ninguno • XON/OFF • RTS/CTS • Manejo automático de las señales V.24	Ninguno
Velocidad de transmisión	Seleccione la velocidad de la transmisión de datos en bits por segundo.	• 110 • 300 • 600 • 1200 • 2400 • 4800 • 9600 • 19200 • 38400 • 57600 • 76800 • 115200	9600
Bits de datos	Seleccione la cantidad de bits en la que se reproducirá un carácter.	• 7 • 8	8
Bits de parada	Seleccione la cantidad de bits de parada que se colocan detrás de cada carácter durante la transferencia y que marcan el fin de un carácter.	• 1 • 2	1

Parámetro	Descripción	Rango de valores	Valor por defecto
Paridad	La secuencia de los bits de datos puede ampliarse en un carácter para incluir el bit de paridad. El valor adicional (0 ó 1) traslada el valor de todos los bits (bits de datos y bit de paridad) a un estado definido. **Sin**: Los datos se envían sin bit de paridad. **Impar**: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es impar con el estado de señal "1". **Par**: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es par con el estado de señal "1". **Cualquiera**: El estado de señal del bit de paridad es irrelevante. La paridad no se comprueba al recibir datos y siempre se ajusta a "0" cuando se envían datos.	• Sin • Impar • Par • Cualquiera	Par

Parámetro	Descripción	Rango de valores	Valor por defecto
Indicador de fin del telegrama de recepción	Cuando se transfieren datos mediante el driver ASCII, el fin del telegrama de recepción se reconoce de tres modos distintos. Aquí puede seleccionar uno de los tres modos de transmisión posibles e indicar los parámetros específicos. **Nota**: Si se agota el tiempo de retardo entre caracteres mientras se reciben datos, la recepción se interrumpe antes de tiempo en los tres modos de operación. El telegrama se deseche, excepto en el modo de operación "Transcurrido el tiempo de retardo entre caracteres". • Transcurrido el tiempo de retardo entre caracteres: El fin de telegrama se reconoce cuando se ha excedido el tiempo de retardo entre caracteres parametrizado. • Recepción del/de los carácter(es) de fin: El final del telegrama se reconoce al recibir el/los carácter(es) de fin definidos. • Recepción de un número determinado de caracteres: El fin de telegrama se reconoce por la longitud de telegrama parametrizada. Todos los telegramas que deben recibirse tienen la misma longitud.	• Transcurrido el tiempo de retardo entre caracteres • Recepción del/de los carácter(es) de fin • Una vez recibido un número de caracteres determinado	Transcurrido el tiempo de retardo entre caracteres
Transcurrido el tiempo de retardo entre caracteres, ms	El intervalo máximo de tiempo que puede transcurrir entre la recepción de dos caracteres. [1]	De 1 a 65535 ms	4 ms
Carácter de fin 1[2]	Para recibir datos con carácter de fin se pueden definir dos caracteres de fin como máximo. Los caracteres de fin seleccionados delimitan la longitud del telegrama.	• Con 7 bits de datos:[3] 1 a $7F_H$ • Con 8 bits de datos:[3] 1 a FF_H	3
Carácter de fin 2[2]	Para recibir datos con carácter de fin se pueden definir dos caracteres de fin como máximo. Los caracteres de fin seleccionados delimitan la longitud del telegrama. Código del segundo carácter de fin, si está activado.	• Con 7 bits de datos:[3] 0 a $7F_H$ • Con 8 bits de datos:[3] 0 a FF_H	0

Parámetro	Descripción	Rango de valores	Valor por defecto
Longitud del telegrama al recibir [4]	Indique la longitud de telegrama cuando deban recibirse datos con una cantidad fija de caracteres. La longitud de telegrama debe corresponderse exactamente con la cantidad de bytes de datos que debe recibir el interlocutor.	1 a 224 bytes	100
Telegramas dinámicos	Para recibir mensajes se puede indicar si sólo debe respaldarse un mensaje o si los mensajes deben respaldarse de forma dinámica. Activando los telegramas dinámicos, el módulo puede respaldar varios mensajes con longitudes diferentes. El búfer es un búfer circular. Si el búfer está lleno se sobrescribe el mensaje más antiguo, a no ser que el parámetro "Impedir sobrescritura del búfer" esté activado. En este caso, se desecha el mensaje más reciente. En ambos casos, una alarma de diagnóstico indica que se han perdido datos.	• Activado • Bloqueado	Activado
Impedir sobrescritura del búfer	Con este parámetro se impide que se sobrescriban los telegramas respaldados cuando el módulo recibe un telegrama nuevo y el búfer de recepción todavía no se ha borrado. De este modo se impide que se pierdan telegramas recibidos anteriormente.	• No • Sí	Sí
Borrar el búfer de recepción del ET 200S 1SI en el arranque	Indique si el búfer de recepción del módulo debe borrarse automáticamente cuando la CPU pasa del estado operativo STOP a RUN (arranque de la CPU). Ello permite asegurarse de que el búfer de recepción del módulo sólo recibe telegramas que se han recibido después de arrancar la CPU.	• No • Sí	Sí

[1] El tiempo de retardo entre caracteres más breve se calcula según la velocidad de transmisión.

[2] Sólo ajustable con carácter de fin como criterio de fin.

[3] Dependiendo de si se parametrizan 7 u 8 bits de datos para la trama de caracteres.

[4] Sólo ajustable con longitud fija de telegrama como criterio de fin.

2.9.3 Parametrización de los drivers para el protocolo 3964(R)

Principio

La tabla siguiente lista los parámetros que pueden ajustarse para el protocolo 3964(R) del módulo interfaz serie.

Tabla 2- 9 Parámetros de driver para el protocolo 3964(R)

Parámetro	Descripción	Rango de valores	Valor por defecto
Alarma de diagnóstico	Indique si el módulo crea una alarma de diagnóstico cuando surge un error grave.	• No • Sí	No
Activar reconocimiento BREAK	Si se produce una ruptura de la línea o no se ha conectado ningún cable de interfaz, el módulo notifica el mensaje de error "Break".	• No • Sí	No
Tipo de interfaz	Indique la interfaz eléctrica que debe emplearse.	• RS-232C • RS-422	RS-232C
Ocupación estándar de la línea de recepción	Indique el preajuste de la línea de recepción en el modo de operación RS422. No en el modo de operación RS232C. La configuración "Nivel invertido" sólo es requerida en caso de repuesto para asegurar la compatibilidad.	R(A) 5V / R(B) 0V (BREAK) R(A) 0V / R(B) 5V Nivel invertido	R(A) 5V / R(B) 0V (BREAK)
Modo de operación del protocolo	Indique si los datos deben enviarse con carácter de comprobación de bloque (BCC) para aumentar la protección de los mismos. El carácter de comprobación de bloque es la paridad longitudinal par (enlace EXOR de todos los bytes de datos) de un bloque enviado o recibido. Si un interlocutor reconoce un carácter de comprobación de bloque al recibir datos, compara el BCC con la paridad longitudinal calculada internamente. Si el carácter de comprobación de bloques es erróneo, se espera 4 segundos (tiempo de espera de bloque) y, a continuación, se repite la transmisión de datos. Si el bloque de datos no puede recibirse tras un número parametrizado de intentos de transmisión o si no se produce otro intento dentro del tiempo de espera de bloque, la recepción se cancela.	• Sin comprobación de bloques • Comprobación de bloques	Comprobación de bloques

Parámetro	Descripción	Rango de valores	Valor por defecto
Velocidad de transmisión	Seleccione la velocidad de la transmisión de datos en bits por segundo.	• 110 • 300 • 600 • 1200 • 2400 • 4800 • 9600 • 19200 • 38400 • 57600 • 76800 • 115200	9600
Bits de datos	Seleccione la cantidad de bits en la que se reproducirá un carácter.	• 7 8	8
Bits de parada	Seleccione la cantidad de bits de parada que se colocan detrás de cada carácter durante la transferencia y que marcan el fin de un carácter.	• 1 • 2	1
Paridad	La secuencia de los bits de datos puede ampliarse en un carácter para incluir el bit de paridad. El valor adicional (0 ó 1) traslada el valor de todos los bits (bits de datos y bit de paridad) a un estado definido. • Sin: Los datos se envían sin bit de paridad. • Impar: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es impar con el estado de señal "1". • Par: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es par con el estado de señal "1". • Cualquiera: El estado de señal del bit de paridad es irrelevante. La paridad no se comprueba al recibir datos y siempre se ajusta a "0" cuando se envían datos.	• Sin • Impar • Par • Cualquiera	Par
Tiempo de retardo entre caracteres (ms)	El intervalo máximo de tiempo que puede transcurrir entre la recepción de dos caracteres. Configure para su aplicación el tiempo de retardo entre caracteres más breve. Tenga en cuenta que el tiempo de retardo entre caracteres debe tener un valor mínimo determinado según la velocidad de transmisión.	20 hasta 655350 ms en escala de 10 ms	220 ms

Parámetro	Descripción	Rango de valores	Valor por defecto
Tiempo de retardo de acuse (ms)	Registre el tiempo máximo que puede transcurrir antes de que se reciba un acuse del interlocutor al establecer e interrumpir la conexión. Tenga en cuenta que el tiempo de retardo de acuse debe tener un valor mínimo determinado según la velocidad de transmisión.	10 hasta 655350 ms en escala de 10 ms	2000 ms (550 ms sin comprobación de bloque)
Intentos de establecimiento	Indique la cantidad de (*n*) intentos para establecer una conexión. (Tras *n* intentos fallidos se cancela la función y aparece el error en la salida STATUS del bloque de función S_SEND.)	1 a 255	6
Intentos de transmisión	Indique el número (*n*) de intentos de transmisión de un telegrama. (Tras *n* intentos fallidos de enviar el telegrama sin errores, la función se interrumpe y el fallo se indica en la salida STATUS del bloque de función S_SEND.) Posibles causas de la interrupción: • Error de paridad • Error de BBC; error de paridad • Parametrización diferente de los interlocutores (p. ej. velocidad de transmisión, paridad, trama de caracteres, carácter de comprobación de bloques, protocolos diferentes)	1 a 255	6
Prioridad	Si ambos interlocutores establecen simultáneamente la comunicación, aquel que tenga menor prioridad retirará su petición de envío en primer lugar. Para la transmisión de datos tiene que asignar una prioridad mayor a un interlocutor y una menor al otro interlocutor.	• Alta • Baja	Baja
Borrar el búfer de recepción del ET 200S 1SI en el arranque	Indique si el búfer de recepción del módulo debe borrarse automáticamente cuando la CPU pasa del estado operativo STOP a RUN (arranque de la CPU). Ello permite asegurarse de que el búfer de recepción del módulo sólo recibe telegramas que se han recibido después de arrancar la CPU.	• No • Sí	Sí

2.9.4 Datos de identificación

Definición

Los datos de identificación son datos almacenados en un módulo que le ayudarán a

- Eliminar fallos de una instalación
- Comprobar la configuración de la instalación
- Localizar las modificaciones del hardware de una planta.

Con los datos de identificación es posible identificar módulos online de forma unívoca. Estos datos están disponibles en los módulos ET 220S 1SI a partir de la ref. 6ES7138-4DFx1-0AE0.

Puede ver los datos identificativos con Sistema de destino > Información del módulo o, como se describe a continuación, con "Leer registro".

Lectura de los datos identificativos

El comando Leer registro permite al usuario acceder de forma selectiva a determinados datos identificativos.

Debajo de cada número de registro aparece la sección de los datos identificativos correspondiente al índice en cuestión.

- Todos los registros que disponen de datos identificativos tienen una longitud de 64 bytes.
- Los registros están configurados de acuerdo con el principio representado en la tabla siguiente.

Tabla 2- 10 Principio de estructura de los registros que contienen datos identificativos

Contenido	Longitud (bytes)	Codificación (hex)
Información de cabecera		
ID de SZL	2	F1 11
Índice	2	00 0x
Longitud de los datos identificativos	2	00 38
Número de bloques con datos identificativos	2	00 01
Datos de identificación		
Índice	2	00 0x
Datos identificativos correspondientes al índice en cuestión (véase la tabla siguiente)	54	

Datos identificativos del módulo ET 200S 1SI

Tabla 2- 11 Datos identificativos del módulo ET 200S 1SI

Datos de identificación	Acceso	Ajuste predeterminado	Explicación
Índice 1 (registro 231/sólo lectura)			
Fabricante	Leer (2 bytes)	00 2A hex (= 42 dec)	Aquí se guarda el nombre del fabricante. (42 dec = Siemens AG)
Denominación del equipo	Leer (20 bytes)	6ES7 138-4DFx1-0AB0	Referencia del módulo x = 0 (ASCII/3964R), 1 (MODBUS/USS)
Número de serie del equipo	Leer (16 bytes)	Aquí se almacena el número de serie del módulo. Ello permite identificar el módulo de forma unívoca.	
Revisión del hardware	Leer (2 bytes)	Informa sobre la versión del módulo.	
Revisión de software	Leer (4 bytes)	Informa sobre la versión de firmware del módulo.	
Número de revisión estadístico	Leer (2 bytes)	–	No soportado
Profile_ID	Leer (2 bytes)	F6 00 hex	Parámetro interno (según PROFIBUS DP)
Profile–specific type	Leer (2 bytes)	00 04 hex (= 4 dec)	Parámetro interno (módulo de comunicación, según PROFIBUS DP)
I&M Version	Leer (2 bytes)	00 00 hex (= 0 dec)	Parámetro interno (según PROFIBUS DP)
I&M supported	Leer (2 bytes)	00 01 hex (= 1 dec)	Parámetro interno (I&M0 e I&M1, según PROFIBUS DP)
Índice 2 (registro 232/lectura y escritura)			
AKZ	Leer/escribir (máx. 32 caracteres)	–	Subdivisión fundamental del módulo
OKZ	Leer/escribir (máx. 22 caracteres)	–	Identificador de ubicación del módulo

2.9.5 Cargar actualizaciones de firmware a posteriori

Descripción

Para ampliar funciones y eliminar los fallos se pueden cargar actualizaciones de firmware en la memoria del sistema operativo del ET 200S 1SI.

La carga a posteriori de actualizaciones de firmware se efectúa desde HW Config.

Firmware básico

El ET 200S 1SI se suministra con un firmware básico.

Requisitos

Para poder cargar actualizaciones de firmware deben cumplirse los siguientes requisitos:

- El ET 200S 1SI debe estar accesible online desde la PG o el PC.
- Los archivos con la nueva versión de firmware deben estar disponibles en el sistema de archivos de su PG/PC.

Cargar el firmware

Proceda del siguiente modo para realizar una actualización de firmware (sólo aplicable si el IM 151 soporta esta función):

1. Abra HW Config y seleccione el módulo ET 200S 1SI que desee actualizar.
2. Elija el comando de menú Sistema de destino > Actualizar firmware.

El procedimiento posterior se describe en la Ayuda en pantalla de STEP 7.

Nota

Para cargar el archivo de firmware para el módulo ET 200S 1SI, ponga la CPU en modo STOP.

Si la actualización es correcta, aparecerá un mensaje de confirmación y el nuevo firmware quedará activado inmediatamente.

Después de la actualización, debe colocarse un adhesivo con la nueva versión de firmware del ET 200S 1SI sobre el adhesivo que indicaba la versión instalada hasta entonces.

Actualización sin éxito

Si la actualización no ha salido bien, el LED SF del módulo parpadea. Repita la actualización. Si resulta imposible ejecutar la actualización correctamente, diríjase al representante de Siemens.

Indicadores LED

Tabla 2- 12 Indicaciones mediante LED durante la carga de una actualización de firmware

Estado	SF	TXD	RXD	Observación	Solución
Actualización de firmware en curso	Encendido	Encendido	Encendido	-	-
Actualización de firmware finalizada	Encendido	apagado	apagado	-	-
ET 200S 1SI sin firmware de módulo	Parpadea (2Hz)	apagado	apagado	Firmware de módulo borrado, la actualización fue interrumpida, es posible ejecutar una actualización de firmware	Recargar el firmware
Error de hardware en la actualización del firmware	Parpadea (2Hz)	Parpadea (2Hz)	Parpadea (2Hz)	Borrado/escritura sin éxito	Desconectar y conectar la tensión de alimentación del módulo y volver a cargar el firmware. Comprobar si el módulo está defectuoso.

Consultar la versión de hardware y de firmware

La versión actual del hardware y del firmware instalados en el ET 200S 1SI puede consultarse en **STEP 7** en el diálogo "Información del módulo". A este cuadro de diálogo se accede:

En el SIMATIC Manager: **Archivo > Abrir > Proyecto > Abrir HW Config > Equipo > Abrir online >** y doble clic sobre el módulo ET 200S 1SI.

2.10 Comunicación a través de bloques de función

2.10.1 Principios de la comunicación a través de bloques de función

Resumen

La comunicación entre CPU, ET 200S 1SI y un interlocutor se realiza a través de los bloques de función y de los protocolos del módulo ET 200S 1SI. (Encontrará información sobre la comunicación con CPUs de otros fabricantes (no S7) en el apartado Principios de los datos de referencia (Página 92).)

Los bloques de función constituyen la interfaz de software entre la CPU y el módulo interfaz serie ET 200S 1SI. Éstos tienen que llamarse cíclicamente desde el programa de usuario.

Establecimiento de la comunicación con la CPU

Cada vez que se arranca la CPU, el módulo ET 200S 1SI recibe los parámetros actuales mediante los servicios del sistema de la CPU. Una vez establecida la conexión entre la CPU y el módulo ET 200S 1SI, debe inicializarse el módulo ET 200S 1SI.

Cada bloque de función tiene un mecanismo de arranque propio. Antes de que puedan gestionarse peticiones activamente debe haber finalizado el mecanismo de arranque correspondiente.

El módulo ET 200S 1SI puede desencadenar una alarma de diagnóstico en la CPU. Para ello, el sistema operativo pone a disposición del usuario 2 bytes de información de alarma. El usuario debe programar la evaluación de la información de alarma (OB82). No se permite llamar los bloques de función en el programa de alarma de proceso o de diagnóstico. En los bloques de función no se bloquean las alarmas.

La conversión del protocolo se realiza en el módulo ET 200S 1SI. De acuerdo con el protocolo seleccionado (procedimiento 3964(R) o driver ASCII), la interfaz del módulo ET 200S 1SI se adapta a la interfaz del interlocutor.

Bloques de función del módulo ET 200S 1SI

El sistema de automatización S7-300 pone a su disposición una serie de bloques de función que lanzan y controlan la comunicación entre la CPU y el módulo interfaz serie ET 200S 1SI en el programa de usuario. La tabla siguiente recoge los FBs utilizados por el módulo ET 200S 1SI.

Tabla 2- 13 Bloques de función del módulo ET 200S 1SI

FB	Nombre	Significado
FB2	S_RCV	El bloque de función S_RCV permite recibir datos de un interlocutor y depositarlos en un bloque de datos.
FB3	S_SEND	El bloque de función S_SEND permite enviar un área entera de un bloque de datos o una parte de la misma a un interlocutor.
FB4	S_VSTAT	El bloque de función S_VSTAT permite leer los estados de señalización en la interfaz RS232C del módulo ET 200S1SI.
FB5	S_VSET	El bloque de función S_VSET permite activar y borrar las salidas de la interfaz RS232C del módulo ET 200S 1SI.
FB6	S_XON	Con el bloque de función S_XON pueden ajustarse otros parámetros si el módulo ha sido parametrizado para el control de flujo XON/XOFF.
FB7	S_RTS	Con el bloque de función S_RTS pueden ajustarse otros parámetros si el módulo ha sido parametrizado para el control de flujo RTS/CTS.
FB8	S_V24	Con el bloque de función S_V24 pueden ajustarse otros parámetros si el módulo ha sido parametrizado para el manejo automático de las señales de V.24.

Nota

Estos bloques de datos de instancia no deben cargarse en la CPU mientras esté activa la comunicación de los bloques SEND y RECEIVE.

Consulte también

Datos técnicos (Página 117)

2.10.2 Bloque de función FB3 S_SEND

FB3 S_SEND: Enviar datos a un interlocutor

El FB S_SEND transmite al módulo ET 200S 1SI un grupo de datos de un bloque de datos especificado por los parámetros DB_NO, DBB_NO y LEN. Para la transmisión de datos, el FB S_SEND se llama en el ciclo de forma estática (sin condiciones) o, **alternativamente,** en un programa controlado por tiempos.

Con flanco positivo en la entrada REQ se inicia la transmisión de los datos. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

El bloque de función FB S_SEND se puede llamar en el ciclo con el estado de señal "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al módulo ET 200S 1SI y el FB S_SEND pasa al estado básico. Los datos que ya ha recibido el módulo ET 200S 1SI se envían al interlocutor. Si hay un estado de señal "1" estático en la entrada R, la transmisión está desactivada.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI con el que debe realizarse la conexión.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de error, en STATUS se indica el número de evento correspondiente (véase el capítulo "Diagnóstico (Página 110)"). Si no se produce ningún fallo, STATUS tiene el valor 0. DONE y ERROR/STATUS se indican también en caso de RESET del FB S_SEND (véase el cronograma). Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Arranque

El parámetro COM_RST del FB S_SEND permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Nota

El bloque de función S_SEND no tiene comprobación de parámetros; si la parametrización es errónea, la CPU puede cambiar al estado STOP.

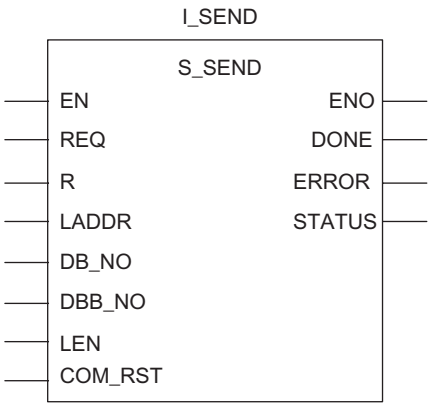
Antes de que el módulo ET 200S 1SI pueda procesar una petición lanzada después de que la CPU pase del estado STOP a RUN, debe haber finalizado el proceso de arranque de la CPU del módulo ET 200S para el FB S_SEND (véase arriba). Una petición iniciada entre tanto no se pierde. Se transfiere al módulo ET 200S 1SI una vez concluida la coordinación de arranque.

Llamada de FB3

Representación AWL

```
CALL      S_SEND, I_SEND
   REQ:      =
   R:        =
   LADDR:    =
   DB_NO:    =
   DBB_NO:   =
   LEN:      =
   DONE:     =
   ERROR:    =
   STATUS:   =
   COM_RST:  =
```

Representación KOP



Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Asignación en el área de datos

El FB S_SEND opera conjuntamente con un DB de instancia I_SEND. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: En caso de error, STATUS == W#16#1Exx, puede consultar más detalles sobre el error en la variable SFCERR (véase el capítulo Diagnóstico (Página 110)). Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Parámetros del FB3 S_SEND

La tabla siguiente lista los parámetros de S_SEND (FB3).

Tabla 2- 14 FB3: Parámetros de S_SEND

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
REQ	INPUT	BOOL	Inicio de petición con flanco positivo	
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI	La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos	Nº de DB de envío: Específico de la CPU (no se permite el valor 0)
DBB_NO	INPUT	INT	Número del byte de datos	0 ≤ DBB_NO ≤ 8190 datos de transmisión a partir de la palabra de datos
LEN	INPUT	INT	Longitud de datos	1 ≤ LEN ≤ 224 indicación en número de bytes
DONE[1]	OUTPUT	BOOL	La petición ha finalizado sin errores	Parámetro STATUS == 16#00
ERROR[1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información sobre el error.

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
STATUS[1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, en el parámetro STATUS figura información sobre el error.
COM_RST	IN_OUT	BOOL	Rearranque del FB	
[1] Tras una petición de envío correcta, los parámetros DONE, ERROR y STATUS están disponibles durante un ciclo de la CPU.				

Cronograma de FB3 S_SEND

La figura siguiente muestra el comportamiento de los parámetros DONE y ERROR según sea el cableado de las entradas REQ y R.

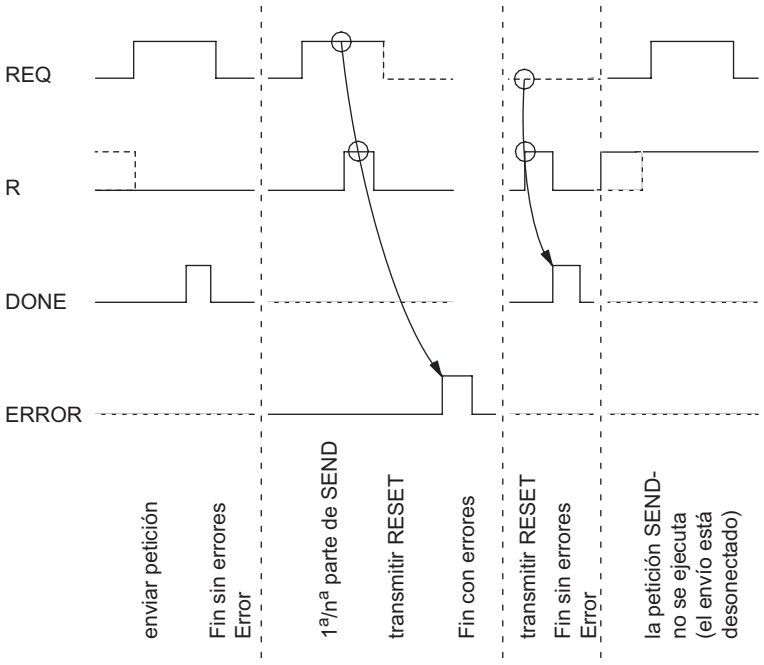


Figura 2-20 Cronograma de FB3 S_SEND

Nota

La entrada REQ está activada por flanco. Es suficiente un flanco positivo en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

2.10.3 Bloque de función FB2 S_RCV

FB S_RCV: Recibir datos de un interlocutor

El FB S_RCV transmite datos de un módulo ET 200S 1SI a un área de datos S7 especificado por los parámetros DB_NO, DBB_NO. Para la transmisión de datos, el FB S_RCV se llama cíclicamente o bien estáticamente (sin condiciones) en un programa controlado por tiempo.

Con el estado de señal (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos del módulo ET 200S 1SI. Una transmisión en curso se puede cancelar con el estado de señal "0" en el parámetro EN_R. La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desconectada mientras exista el estado de señal "0" en el parámetro EN_R. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Si el bloque de función detecta el estado de señal "1" en el parámetro R, se cancela la petición actual de transmisión y el FB S_RCV pasa al estado inicial. La recepción estará desactivada mientras el parámetro R tenga el estado lógico "1". Cuando el estado de señal vuelve a ser "0", el telegrama interrumpido se vuelve a recibir desde el principio.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI con el que debe realizarse la conexión.

La salida NDR indica "Petición terminada sin error/Datos aceptados" (leídos todos los datos). ERROR indica que se ha producido un error. En caso de producirse un error, el número correspondiente se indica en STATUS. Si el búfer de recepción está lleno en más de 2/3 partes, STATUS contiene una advertencia después de cada llamada de S_RCV. Si no hay errores ni advertencias, STATUS tiene el valor "0".

NDR y ERROR/STATUS también se emiten cuando se restaura el FB S_RCV (parámetro LEN == 16#00) (véase el cronograma). Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Arranque

El parámetro COM_RST del FB S_RCV permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Nota

El bloque de función S_RCV no tiene comprobación de parámetros; si la parametrización es errónea, la CPU puede cambiar al estado STOP.

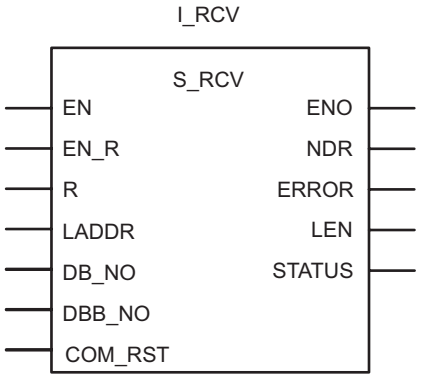
Antes de que el módulo ET 200S 1SI pueda recibir una petición lanzada después de que la CPU pase del estado STOP a RUN, debe haber finalizado el proceso de arranque de la CPU del módulo ET 200S para el FB S_RCV.

Llamada del FB 2

Representación AWL

```
CALL            S_RCV, I_RCV
      EN_R:      =
      R:         =
      LADDR:     =
      DB_NO:     =
      DBB_NO:    =
      NDR:       =
      ERROR:     =
      LEN:       =
      STATUS:    =
      COM_RST:   =
```

Representación KOP



Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Asignación en el área de datos

El FB S_RCV opera conjuntamente con un DB de instancia I_RCV. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: En caso de error (STATUS == W#16#1Exx) se pueden consultar más detalles sobre el error en la variable SFCERR . Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Parámetros del FB2 S_RCV

La tabla siguiente lista los parámetros de S_RCV (FBs).

Tabla 2- 15 FB2: Parámetros de S_RCV

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
EN_R	INPUT	BOOL	Habilitar lectura de datos	
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Bloqueo de recepción.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI	La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos	N.º DB de recepción: Específico de la CPU. No se permite el valor 0.
DBB_NO	INPUT	INT	Número del byte de datos	0 ≤ DBB_NO ≤ 8190 datos de recepción a partir de la palabra de datos
NDR[1]	OUTPUT	BOOL	La petición ha finalizado sin errores, datos transferidos	Parámetro STATUS == 16#00
ERROR[1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información sobre el error.
LEN[1]	OUTPUT	INT	Longitud del telegrama recibido	1 ≤ LEN ≤ 224 indicación en número de bytes
STATUS[1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, en el parámetro STATUS figura información sobre el error.
COM_RST	IN_OUT	BOOL	Rearranque del FB	

[1] Tras una petición de recepción correcta, los parámetros NDR, ERROR, LEN y STATUS están disponibles durante **un** ciclo de la CPU.

Cronograma del FB2 S_RCV

La figura siguiente muestra el comportamiento de los parámetros NDR, LEN y ERROR según sea el cableado de las entradas EN_R y R.

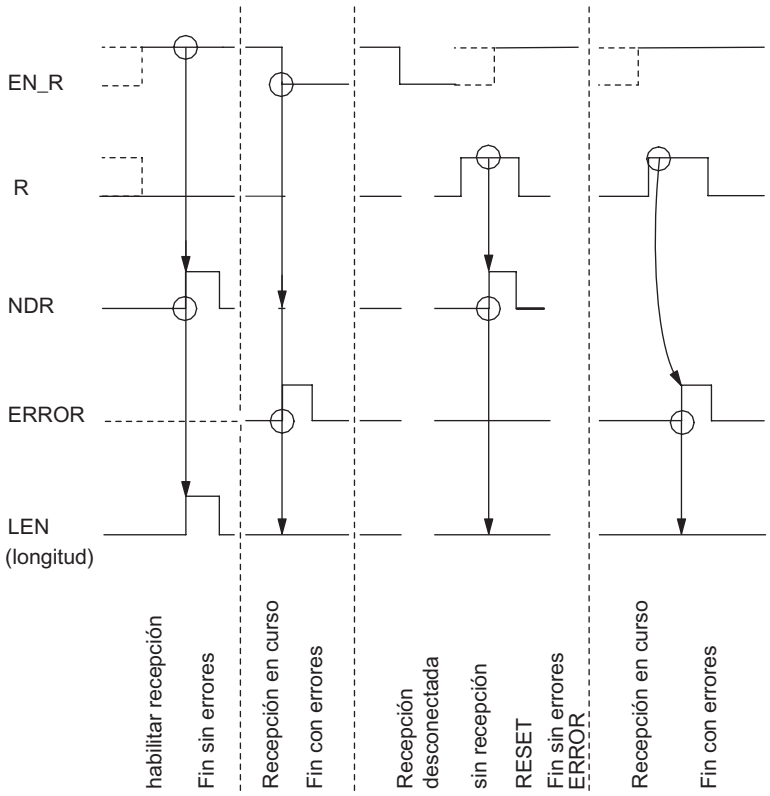


Figura 2-21 Cronograma del FB2 S_RCV

Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

2.10.4 Funciones para parametrizar opciones para el control del flujo de datos

Principio

Si utiliza el módulo interfaz serie ET 200S 1SI con una CPU S7 y configura el módulo con la configuración de hardware de STEP 7, es posible seleccionar un método de control del flujo de datos entre las opciones siguientes:

- Sin
- XON/XOFF
- RTS/CTS
- Manejo automático de las señales V.24

Para cada una de estas opciones pueden configurarse parámetros adicionales. Estos parámetros adicionales adoptan valores predeterminados, que son valores típicos adecuados para la mayoría de las aplicaciones. De todas formas, estos parámetros pueden modificarse mediante el programa de usuario y los siguientes bloques de función.

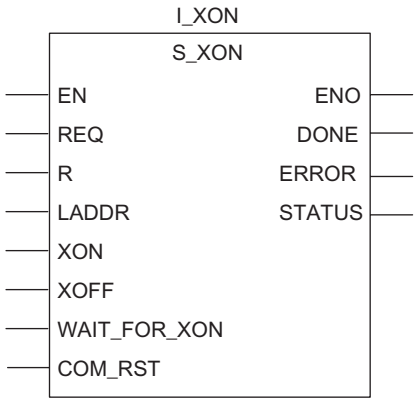
FB6 S_XON: Ajustar los caracteres para XON/XOFF

Con el bloque de función S_XON pueden ajustarse parámetros adicionales (véanse los parámetros del FB6) si el módulo ha sido parametrizado para el control de flujo XON/XOFF.

Representación AWL

```
CALL                    S_XON, I_XON
   REQ:                  =
   R:                    =
   LADDR:                =
   XON:                  =
   XOFF:                 =
   WAIT_FOR_XON:         =
   DONE:                 =
   ERROR:                =
   STATUS:               =
   COM_RST:              =
```

Representación KOP



Asignación en el área de datos

El FB S_XON opera conjuntamente con un DB de instancia I_XON. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: En caso de error (STATUS == W#16#1Exx) se pueden consultar más detalles sobre el error en la variable SFCERR . Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Parámetros del FB6

La tabla siguiente lista los parámetros del FB6.

Tabla 2- 16 FB6: Parámetros de S_XON

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación	Valor predeterminado
REQ	INPUT	BOOL	Inicio de petición con flanco positivo		
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Envío bloqueado.	
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI	La dirección inicial se toma de STEP 7.	
XON	INPUT	BYTE	Carácter XON	0 a $7F_H$ (7 bits de datos) 0 a FF_H (8 bits de datos)	11 (DC1)
XOFF	INPUT	BYTE	Carácter XOFF	0 a $7F_H$ (7 bits de datos) 0 a FF_H (8 bits de datos)	13 (DC3)
WAIT_FOR_XON	INPUT	TIME	Tiempo de espera para XON tras XOFF	20 ms a 10 min 55 s 350ms	2 s
DONE[1]	OUTPUT	BOOL	La petición ha finalizado sin errores	Parámetro STATUS == 16#00	
ERROR[1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información sobre el error.	
STATUS[1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, en el parámetro STATUS figura información sobre el error.	
COM_RST	IN_OUT	BOOL	Rearranque del FB		

[1] Tras una petición correcta, los parámetros DONE, ERROR y STATUS están disponibles durante **un** ciclo de la CPU.

Arranque

El parámetro COM_RST del FB S_XON permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

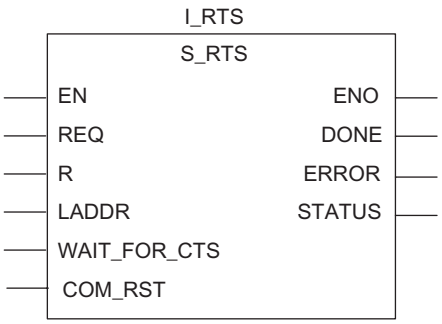
FB7 S_RTS: Configurar los parámetros para RTS/CTS

Con el bloque de función S_RTS pueden ajustarse parámetros adicionales (véanse los parámetros del FB7) si el módulo ha sido parametrizado para el control de flujo RTS/CTS.

Representación AWL

```
CALL                    S_RTS, I_RTS
   REQ:                  =
   R:                    =
   LADDR:                =
   WAIT_FOR_CTS:         =
   DONE:                 =
   ERROR:                =
   STATUS:               =
   COM_RST:              =
```

Representación KOP



Asignación en el área de datos

El FB S_RTS opera conjuntamente con un DB de instancia I_RTS. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: En caso de error (STATUS == W#16#1Exx) se pueden consultar más detalles sobre el error en la variable SFCERR . Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Parámetros del FB7

La tabla siguiente lista los parámetros del FB7.

Tabla 2- 17 FB7: Parámetros de S_RTS

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación	Valor predeterminado
REQ	INPUT	BOOL	Inicio de petición con flanco positivo		
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Envío bloqueado.	
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S SI	La dirección inicial se toma de STEP 7.	
WAIT_FOR_CTS	INPUT	TIME	Tiempo de espera para CTS = ON	20 ms a 10 min 55 s 350ms	2 s
DONE[1]	OUTPUT	BOOL	La petición ha finalizado sin errores	Parámetro STATUS == 16#00	
ERROR[1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información sobre el error.	
STATUS[1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, en el parámetro STATUS figura información sobre el error.	
COM_RST	IN_OUT	BOOL	Rearranque del FB		
[1] Tras una petición correcta, los parámetros DONE, ERROR y STATUS están disponibles durante **un** ciclo de la CPU.					

Arranque

El parámetro COM_RST del FB S_RST permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

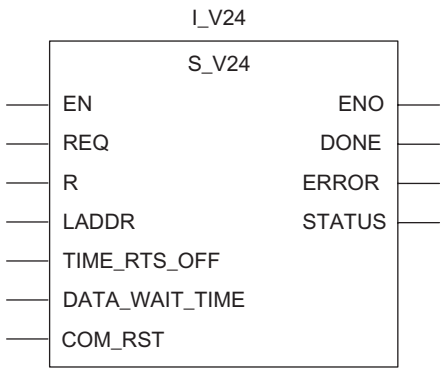
FB8 S_V24: Configurar los parámetros para el manejo automático de las señales cualificadoras RS-232C

El bloque de función S_V24 permite configurar parámetros adicionales (véanse los parámetros del FB8) si el módulo está parametrizado para el manejo automático de las señales cualificadoras RS232C.

Representación AWL

```
CALL                    S_V24, I_V24
   REQ:                  =
   R:                    =
   LADDR:                =
   TIME_RTS_OFF:         =
   DATA_WAIT_TIME:       =
   DONE:                 =
   ERROR:                =
   STATUS:               =
   COM_RST:              =
```

Representación KOP



Asignación en el área de datos

El FB P_V24 opera conjuntamente con un DB de instancia I_V24. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: En caso de error (STATUS == W#16#1Exx) se pueden consultar más detalles sobre el error en la variable SFCERR . Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Parámetros del FB8

La tabla siguiente lista los parámetros del FB8.

Tabla 2- 18 FB8: Parámetros de S_V24

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación	Valor pre-determinado
REQ	INPUT	BOOL	Inicio de petición con flanco positivo		
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Envío bloqueado.	
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI	La dirección inicial se toma de STEP 7.	
TIME_RTS_OFF	INPUT	TIME	Tiempo que debe transcurrir después de la transmisión antes de que se desconecte RTS.	0 ms a 10 min 55 s 350ms	10 ms
DATA_WAIT_TIME	INPUT	TIME	Tiempo que se espera para que el interlocutor active CTS = ON después de activar RTS.	0 ms a 10 min 55 s 350ms	10 ms
DONE[1]	OUTPUT	BOOL	La petición ha finalizado sin errores	Parámetro STATUS == 16#00	
ERROR[1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información sobre el error.	
STATUS[1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, en el parámetro STATUS figura información sobre el error.	
COM_RST	IN_OUT	BOOL	Rearranque del FB		
[1] Tras una petición correcta, los parámetros DONE, ERROR y STATUS están disponibles durante **un** ciclo de la CPU.					

Arranque

El parámetro COM_RST del FB S_V24 permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

2.10.5 Lectura y forzado de señales cualificadoras de RS-232C

Principio

Para leer y forzar las señales cualificadoras de RS232C se dispone de los bloques de función FB4 S_VSTAT, que verifica los estados de las interfaces, y FB S_VSET, que activa/desactiva las salidas de las interfaces.

FB4 S_VSTAT: Comprobar el estado de interfaz del módulo ET 200S 1SI

El FB S_VSTAT lee las señales cualificadoras RS232C del módulo ET 200S 1SI y las pone a disposición del usuario en los parámetros de bloque. Para la transmisión de datos, el FB S_VSTAT se llama de forma estática (sin condiciones) en el ciclo o, alternativamente, en un programa controlado por tiempos.

Las señales cualificadoras RS 232C se actualizan cada vez que se llama la función (sondeo cíclico).

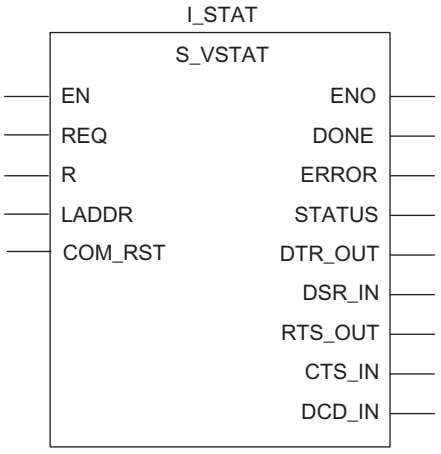
En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI con el que debe realizarse la conexión.

Llamada de FB4

Representación AWL

```
CALL    S_VSTAT, I_STAT
   REQ:      =
   R:        =
   LADDR:    =
   DONE:     =
   ERROR:    =
   STATUS:   =
   DTR_OUT:  =
   DSR_IN:   =
   RTS_OUT:  =
   CTS_IN:   =
   DCD_IN:   =
   COM_RST:  =
```

Representación KOP



Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Asignación en el área de datos

El FB S_VSTAT trabaja con el DB de instancia I_STAT. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Para detectar un cambio de señal es necesario una duración de impulso mínima. Las magnitudes decisivas son el tiempo de ciclo de la CPU, el tiempo de actualización en el módulo ET 200S 1SI y el tiempo de respuesta del interlocutor.

Parámetros del FB4 V24_STAT

La tabla siguiente lista los parámetros del bloque de función S_VSTAT (FB4).

Tabla 2- 19 FB4: Parámetros de V24_STAT

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
REQ	INPUT	BOOL	Inicio de petición con flanco positivo	
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI	La dirección inicial se toma de STEP 7.
DONE [1]	OUTPUT	BOOL	Indica que el FB ha finalizado	(Salida ET 200S 1SI)
ERROR [1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información sobre el error.
STATUS [1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, en el parámetro STATUS figura información sobre el error.
DTR_OUT [1]	OUTPUT	BOOL	Data terminal ready, ET 200S 1SI listo para funcionar.	(Salida ET 200S 1SI)
DSR_IN [1]	OUTPUT	BOOL	Data set ready, interlocutor listo para funcionar.	(Entrada ET 200S 1SI)

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
RTS_OUT [1]	OUTPUT	BOOL	Request to send, ET 200S 1SI listo para enviar.	(Salida ET 200S 1SI)
CTS_IN [1]	OUTPUT	BOOL	Clear to send, el interlocutor puede recibir datos del módulo ET 200S 1SI (respuesta a RTS = ON del ET 200S 1SI)	(Entrada ET 200S 1SI)
DCD_IN [1]	OUTPUT	BOOL	Data carrier detect, nivel de señal de recepción	(Entrada ET 200S 1SI)
COM_RST	IN_OUT	BOOL	Rearranque del FB	
[1] Tras una petición correcta, estos parámetros están disponibles durante **un** ciclo de la CPU.				

Arranque

El parámetro COM_RST del FB S_VSTAT permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

FB5 S_VSET: Activar/desactivar las salidas de las interfaces del módulo ET 200S 1SI

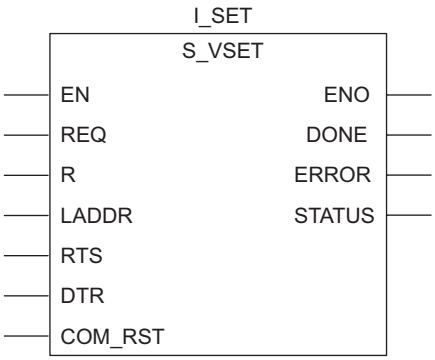
Las salidas de las interfaces pueden activarse y desactivarse mediante las correspondientes entradas de parámetros del FB S_VSET. El bloque de función FB S_VSET se llama de forma estática (sin condiciones) en el ciclo o, alternativamente, en un programa controlado por tiempos.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI con el que debe realizarse la conexión.

Representación AWL

```
CALL        S_VSET, I_SET
   REQ       =
   R         =
   LADDR:    =
   RTS:      =
   DTR:      =
   DONE:     =
   ERROR:    =
   STATUS:   =
   COM_RST:  =
```

Representación KOP



Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Asignación en el área de datos

El FB S_VSET trabaja con un DB de instancia I_SET. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Parámetros del FB5 S_VSET

La tabla siguiente lista los parámetros del bloque de función S_VSET (FB5).

Tabla 2- 20 FB5: Parámetros de S_VSET

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
REQ	INPUT	BOOL	Inicio de petición con flanco positivo	
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI	La dirección inicial se toma de STEP 7.
RTS	INPUT	BOOL	Request to send, ET 200S 1SI listo para enviar.	(forzar salida ET 200S 1SI)
DTR	INPUT	BOOL	Data terminal ready, ET 200S 1SI listo para funcionar.	(forzar salida ET 200S 1SI)
DONE [1]	OUTPUT	BOOL	Indica que el FB ha finalizado	(Salida ET 200S 1SI)
ERROR [1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información sobre el error.
STATUS [1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, en el parámetro STATUS figura información sobre el error.
COM_RST	IN_OUT	BOOL	Rearranque del FB	
[1] Tras una petición correcta, estos parámetros están disponibles durante **un** ciclo de la CPU.				

Arranque

El parámetro COM_RST del FB S_VSET permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

2.11 Propiedades de arranque y estados operativos

Estados operativos del módulo interfaz serie ET 200S 1SI

El módulo ET 200S 1SI dispone de los siguientes estados operativos:

- **STOP:** En el estado STOP del módulo ET 200S 1SI no hay ningún driver de protocolo activo y la CPU acusa negativamente todas las peticiones de envío y recepción. El módulo ET 200S 1SI se mantiene en estado operativo STOP hasta que se ha eliminado la causa del STOP (p.ej. rotura de hilo o parámetro no válido).
- **Reparametrización:** Si se vuelve a parametrizar el módulo ET 200S 1SI, se inicializa el driver de protocolo. Durante la reparametrización, el LED SF permanece encendido.

  No es posible ni enviar ni recibir, los telegramas de envío y recepción guardados en el módulo ET 200S 1SI se pierden a causa del reinicio del driver. La comunicación entre el módulo ET 200S 1SI y la CPU se inicia de nuevo (los telegramas en curso se cancelan).

  Al final de la reparametrización, el módulo ET 200S 1SI está en estado operativo RUN y está listo para enviar y recibir.
- **RUN:** El módulo ET 200S 1SI procesa las peticiones de envío de la CPU. La CPU pone a disposición para su recogida los telegramas recibidos por el interlocutor.

Propiedades de arranque del módulo ET 200S 1SI

El arranque consta de dos fases:

- **Inicialización:** En cuanto el módulo ET 200S 1SI recibe suministro de tensión, se inicializa la interfaz serie, que espera a datos de parametrización de la CPU.
- **Parametrización:** Durante la parametrización, el módulo ET 200S 1SI recibe los parámetros del módulo asignados con STEP 7 al slot actual.

Comportamiento del módulo ET 200S 1SI cuando la CPU cambia de estado operativo

Una vez que el módulo ET 200S 1SI ha arrancado se intercambian todos los datos entre la CPU y el ET 200S 1SI mediante los bloques de función.

- **STOP de la CPU:** En el estado operativo STOP de la CPU no es posible la comunicación a través de PROFIBUS. Si hay una transmisión de datos en curso entre el módulo y la CPU, tanto si es una petición de envío como de recepción, ésta se cancela y la conexión reinicia.

  El intercambio de datos en la interfaz RS232C del módulo ET 200S 1SI continúa con el driver ASCII en la parametrización sin control de flujo, es decir, la petición de envío en curso se termina. Los telegramas de recepción se van recibiendo en el caso del driver ASCII hasta que se llene el búfer de recepción.

- **Arranque de la CPU:** En el arranque, la CPU transmite parámetros al módulo ET 200S 1SI.

  Con la parametrización adecuada se puede borrar automáticamente el búfer de recepción del ET 200 S 1SI al arrancar la CPU.

- **RUN de la CPU:** En el estado operativo RUN de la CPU son posibles los modos de envío y recepción sin restricciones. En las primeras ejecuciones de FB después de reiniciar la CPU se sincronizan el módulo ET 200S 1SI y los FBs correspondientes. Sólo entonces se ejecuta un nuevo FB S_SEND S_RCV.

Particularidades al enviar telegramas

Los telegramas sólo pueden enviarse en estado operativo RUN de la CPU.

Si durante la transmisión de datos de la CPU al módulo la CPU pasa al estado operativo STOP, el FB S_SEND comunica el error (05) 02_H después del rearranque. Para evitar este comportamiento, el programa de usuario puede llamar el FB S_SEND con la entrada RESET desde el OB de arranque.

Nota

El módulo ET 200S 1SI no envía datos al interlocutor hasta que ha recibido todos los datos de la CPU.

Particularidades al recibir telegramas

Con STEP 7 puede parametrizarse "Borrar búfer de recepción del módulo en el arranque = sí/no".

- Si se ha parametrizado "sí", el búfer de recepción del módulo ET 200S 1SI se borra automáticamente cuando la CPU pasa de STOP a RUN.
- Si se ha parametrizado "no", en el búfer de recepción del módulo ET 200S 1SI se respaldan tantos telegramas como se haya parametrizado.

Si durante la transmisión de datos de la CPU al módulo ET 200S 1SI la CPU pasa al estado operativo STOP, el FB comunica el error (05) 02_H después del rearranque. Para evitar este comportamiento, el programa de usuario puede llamar el FB S_SEND con la entrada RESET desde el OB de arranque. En caso de "Borrar búfer de recepción de ET 200S 1SI en el arranque = no", el módulo transfiere de nuevo el telegrama a la CPU.

Búfer dinámico de telegramas

Elija si desea memorizar un solo telegrama o si prefiere memorizarlos dinámicamente. Si selecciona la casilla de verificación "activar", el módulo puede memorizar distintos telegramas de diferente longitud. El búfer es un búfer anular. Cuando se llena, se borra el mensaje más antiguo, a menos que haya activado “No sobreescribir búfer de telegrama". En este caso, se desecha el mensaje más reciente. Cuando se sobreescribe un mensaje se dispara una alarma de diagnóstico que muestra la pérdida de datos.

2.12 Datos de referencia para otros maestros como S7-PROFIBUS

2.12.1 Principios de los datos de referencia

Intercambio de datos entre el maestro y el módulo ET 200S 1SI

El módulo ET 200S 1SI está configurado para transferencias de datos de 4, 8 ó 32 bytes, entrada o salida, con coherencia en toda la longitud. El módulo ET 200S 1SI utiliza la memoria de entrada y salida de 4, 8 ó 32 bytes para transmitir datos a la CPU y recibirlos de ésta mediante medios de transmisión de PROFIBUSDP.

De este modo, la CPU puede escribir datos en todo momento en las entradas y salidas, además de leer datos de las entradas y salidas:

- La CPU transmite una petición al módulo ET 200S 1SI en el primer byte de la memoria de salida del módulo.
- El módulo ET 200S 1SI acepta la petición transmitiendo el código de petición a la memoria de entrada.
- La CPU intercambia datos mediante segmentos de 3, 7 ó 31 bytes (tantos segmentos como sean necesarios de acuerdo con el tamaño de E/S) hasta que se han transferido todos los datos de la petición.

El primer byte del segmento es un byte de coordinación que sirve para sincronizar la transferencia del segmento correspondiente entre la CPU y el módulo ET 200S 1SI (véase la siguiente figura). Los demás bytes de la memoria E/S contienen los datos de la petición.

La CPU transfiere datos al módulo ET 200S 1SI del modo siguiente:

Byte	Contenido
0	Byte de coordinación
1	Byte de datos 0
2	Byte de datos 1
• • • •	• • • •
N	Byte de datos n

El módulo ET 200S 1SI transmite los datos a la CPU del modo siguiente:

Byte	Contenido
0	Byte de coordinación
1	Byte de datos 0
2	Byte de datos 1
• • • •	• • • •
N	Byte de datos n

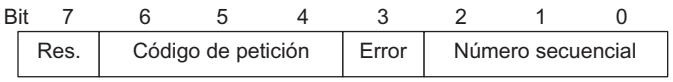
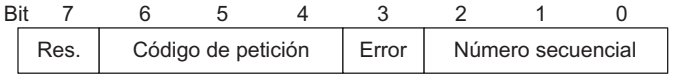
n = 3, 7 ó 31, según la variante de módulo seleccionada en la configuración

Figura 2-22 Intercambio de datos entre la CPU y el módulo ET 200S 1SI

Descripción del byte de coordinación

La tabla siguiente describe el contenido del byte de coordinación (byte 0), que sincroniza la transmisión de datos entre la CPU y el módulo interfaz serie ET 200S 1SI.

Tabla 2- 21 Contenido del byte de coordinación 0 para la transmisión de datos

Segmento de byte	Descripción
Byte de petición escrito por la CPU	
Bit 7	Reservado para aplicaciones especiales del FB S_SEND. Para evaluaciones del byte de coordinación debe ocultarse este bit.
Código de petición	Activado por la CPU para iniciar una petición.
Número secuencial	**Enviar petición:** La CPU lo aumenta en 1 cuando envía otro segmento al módulo ET 200S 1SI...*o bien* **Recibir petición:** El byte de entrada 0 de la CPU lo aplica cuando la CPU recibe un nuevo segmento del módulo interfaz serie en el orden correcto. Muestra el último número secuencial válido cuando el bit de error está activado. (El valor pasa de 1 a 7).
Error	La CPU lo activa para mostrar que un segmento no se ha recibido en el orden correcto. El campo Número secuencial muestra el último número secuencial válido.
Byte de petición escrito por el módulo ET 200S 1SI	
Bit 7	Reservado para aplicaciones especiales del FB S_SEND. Para evaluaciones del byte de coordinación debe ocultarse este bit.
Código de petición	El módulo ET 200S 1SI lo aplica para acusar que se ha recibido la petición.

Segmento de byte	Descripción
Número secuencial	**Enviar petición:** El byte de salida 0 del módulo lo aplica cuando el módulo recibe un nuevo segmento de la CPU en el orden correcto. Muestra el último número secuencial válido cuando el bit de error está activado. **Recibir petición:** El módulo lo aumenta en 1 cuando envía otro segmento a la CPU. (El valor pasa de 1 a 7).
Error	El emisor supervisa el bit de error del receptor en relación a una transacción segmentada. Si el bit de error está activado: • **Emisor CPU** (petición de envío): La CPU vuelve a enviar los segmentos, comenzando por el segmento siguiente al número indicado por el receptor. • **Emisor módulo** (petición de recepción): El módulo 1SI interrumpe la transmisión del telegrama Rx al usuario con el mensaje de error 0x0551 en la palabra de estado. El módulo espera al acuse del mensaje de error (Idle). Una vez finalizada la secuencia de error en curso, el telegrama Rx interrumpido se vuelve a notificar al usuario o bien se pone a disposición.

Definiciones de los códigos de petición

La tabla siguiente lista las peticiones de acuerdo con la asignación de los bits 4 a 6 en el byte de coordinación 0.

Tabla 2- 22 Códigos de petición

Bits 6 5 4	Valor hex.	Definición
0 0 0	0_H	Estado de reposo
0 0 1	1_H	Enviar
0 1 0	2_H	Recibir
0 1 1	3_H	Leer estado de señal V.24
1 0 0	4_H	Escribir señales V.24
1 0 1	5_H	Transferir parámetros: Esta petición permite configurar otros parámetros, que no aparecen en el archivo GSD.
1 1 0	6_H	Reservado
1 1 1	7_H	Acuse de fin de petición

Reglas para escribir códigos de petición

Las reglas siguientes sirven para escribir códigos de petición en el byte de coordinación, lo que permite a la CPU y al módulo ET 200S 1SI sincronizar las transferencias de datos:

- Antes de que el programa de usuario de la CPU pueda escribir un código de petición en el byte de coordinación de salida, tiene que ver un código de reposo del byte de coordinación de entrada del módulo ET 200S 1SI.
- Antes de que el programa de usuario de la CPU pueda escribir el primer segmento en el byte de salida 1..n, tiene que ver el código de acuse de la petición (es decir, el código de petición aceptado) en el byte de coordinación de entrada del módulo.
- Si el programa de usuario ve otros códigos de acuse de petición diferentes a los enviados por el programa, no puede escribir en el byte de salida 0..n hasta que ha visto de nuevo un código de reposo del byte de coordinación de entrada del módulo ET 200S 1SI.

  Esta situación puede producirse, por ejemplo, cuando se ejecutan dos peticiones separadas en el mismo ciclo, ambas peticiones ven el código de reposo y ambas escriben un código de petición distinto en el byte de salida. A causa del ciclo asíncrono entre el ciclo de la CPU y el ciclo de PROFIBUS DP, no se garantiza que la petición llegue primero al módulo. Por ello, cada petición debe poder esperar el fin de la otra petición antes de ser procesada.

Estado de recepción del módulo 1SI

El módulo 1SI muestra su estado de recepción siempre que se encuentra en estado de reposo (byte de acuse de petición 0 = 00H). El estado de recepción está guardado en los bytes 1 y 2.

Estado	Significado
0000_H	No hay ningún mensaje recibido disponible
0001_H	Hay disponible un mensaje o un telegrama de recepción recibido
$0B01_H$	El búfer de recepción está lleno en más de 2/3 partes

Definiciones de las palabras de estado

En los ejemplos de transferencias de datos de las páginas siguientes, el módulo ET 200S 1SI utiliza los bytes 1 y 2 para el mensaje de estado cuando responde a la CPU. La tabla "Mensajes de diagnóstico del parámetro STATUS" lista las palabras de estado y las definiciones.

Orden de los bytes en la palabra

Al transmitir datos entre la CPU y el módulo ET 200S 1SI, en todas las palabras de 16 bits (p. ej. estado y longitud) primero se envía el byte de mayor valor.

Estado de recepción del módulo 1SI

El estado del búfer de recepción del módulo 1SI se muestra al usuario siempre que el módulo se encuentra en estado de reposo (byte de acuse de petición 0 = 00_H). En ese caso, el estado está guardado en los bytes 1+2.

Estado	Significado
0000H	No hay ningún mensaje recibido disponible
0001H	Hay disponible un mensaje o un telegrama de recepción recibido
0B01H	El búfer de recepción está lleno en más de sus 2/3 partes.

2.12.2 Ejemplo del proceso de transmisión de datos de la CPU al módulo

Ejemplo de proceso

La tabla siguiente muestra un ejemplo de una CPU que envía un mensaje con los 22 primeros caracteres del alfabeto. La memoria E/S es de 8 bytes. El ciclo DP es prácticamente igual al ciclo de la CPU, de modo que se produce un tiempo de latencia de un ciclo cuando el módulo responde con el número secuencial.

Tabla 2- 23 Ejemplo de proceso al enviar

<table>
<tr><th>Ciclo de la CPU</th><th>CPU escribe en ET 200S 1SI</th><th>CPU lee del ET 200S 1SI</th></tr>
<tr><td rowspan="4">1.</td><td colspan="2">El programa de usuario ve el siguiente código de reposo del módulo:
Byte 0 1 2 3 4 5 6 7
←
00_H | nnnn_H | xx_H | xx_H | xx_H | xx_H | xx_H
Acus. pet. | Estado | irrelevante</td></tr>
<tr><td colspan="2">La CPU escribe petición para enviar:
Byte 0 1 2 3 4 5 6 7
10_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H →
Petición | irrelevante</td></tr>
<tr><td rowspan="2">2.</td><td colspan="2">El programa de usuario todavía lee el código de reposo del módulo:
←
00_H | nnnn_H | xx_H | xx_H | xx_H | xx_H | xx_H
Acus. pet. | Estado | irrelevante</td></tr>
<tr><td colspan="2">La CPU repite petición para enviar:
10_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H →
Petición | irrelevante</td></tr>
<tr><td rowspan="2">3.</td><td colspan="2">El programa de usuario lee la respuesta siguiente del módulo:
←
10_H | nnnn_H | xx_H | xx_H | xx_H | xx_H | xx_H
Acus. pet. | Estado | irrelevante</td></tr>
<tr><td colspan="2">La CPU envía el 1er segmento:
11_H | 0016_H | ‘a’ | ‘b’ | ‘c’ | ‘d’ | ‘e’ →
Petición | Longitud de envío | Datos</td></tr>
</table>

Ciclo de la CPU	CPU escribe en ET 200S 1SI	CPU lee del ET 200S 1SI
4.	El programa de usuario lee la respuesta siguiente del módulo:	
	←	

10_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H
Acus.pet.	Irrelevante						

La CPU repite el 1er segmento:							
11_H	'f'	'g'	'h'	'i'	'j'	'k'	'l'
Petición	Datos						

→

5.	El programa de usuario lee la respuesta siguiente del módulo:						
	←						
11_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H
Acus.pet.	Irrelevante						
La CPU envía el 2º segmento, ya que no se mostró ningún error y el proceso es correcto:							
12_H	'm'	'n'	'o'	'p'	'q'	'r'	's'
Petición	Datos						

→

6.	El programa de usuario lee la respuesta siguiente del módulo:						
	←						
12_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H
Acus.pet.	Irrelevante						
La CPU envía el 3er segmento, ya que no se mostró ningún error y el proceso es correcto:							
13_H	't'	'u'	'v'	xx_H	xx_H	xx_H	xx_H
Petición	Datos			irrelevante			

→

7.	El programa de usuario lee la respuesta siguiente del módulo:						
	←						
13_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H
Acus.pet.	Irrelevante						
La CPU envía el 4º segmento, ya que no se mostró ningún error y el proceso es correcto:							
14_H	't'	'u'	'v'	xx_H	xx_H	xx_H	xx_H
Petición	Datos			irrelevante			

→

8.	El programa de usuario lee la respuesta siguiente del módulo:						
	←						
13_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H	xx_H
Acus.pet.	Irrelevante						

<table>
<tr><th>Ciclo de la CPU</th><th>CPU escribe en ET 200S 1SI</th><th>CPU lee del ET 200S 1SI</th></tr>
<tr><td></td><td colspan="2">La CPU espera el acuse en el 4º segmento:
<table><tr><td>14_H</td><td>‘t’</td><td>‘u’</td><td>‘v’</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Petición</td><td colspan="3">Datos</td><td colspan="4">irrelevante</td></tr></table>→</td></tr>
<tr><td>9.</td><td colspan="2">El programa de usuario lee la respuesta siguiente del módulo:
←<table><tr><td>14_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Acus.pet.</td><td colspan="7">Irrelevante</td></tr></table></td></tr>
<tr><td></td><td colspan="2">La CPU no envía nada nuevo (las salidas se mantienen iguales) y espera el último acuse del módulo; se indica que el mensaje se ha enviado al interlocutor:
<table><tr><td>14_H</td><td>‘t’</td><td>‘u’</td><td>‘v’</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Petición</td><td colspan="3">Datos</td><td colspan="4">irrelevante</td></tr></table>→</td></tr>
<tr><td>n.</td><td colspan="2">Algunos ciclos de CPU más tarde, el programa de usuario ve la respuesta siguiente del módulo:
←<table><tr><td>74_H</td><td>$nnnn_H$</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Acus. pet.</td><td>Estado</td><td colspan="5">irrelevante</td></tr></table></td></tr>
<tr><td>—</td><td colspan="2">La CPU escribe el código de reposo en la petición y la finaliza.</td></tr>
</table>

2.12.3 Ejemplo de proceso al recibir datos del módulo en la CPU

Ejemplo de proceso

La tabla siguiente muestra un ejemplo de cómo la CPU recibe un mensaje del módulo interfaz serie. La memoria E/S es de 8 bytes. El ciclo DP es más breve que el de la CPU, de modo que no se produce ningún tiempo de latencia en el módulo.

Tabla 2- 24 Ejemplo de proceso al recibir

<table>
<tr><th>Ciclo de la CPU</th><th>CPU escribe en ET 200S 1SI</th><th>CPU lee del ET 200S 1SI</th></tr>
<tr><td>n</td><td>El programa de usuario lee el código de reposo del módulo en varios ciclos, hasta que el estado muestra que hay un mensaje recibido disponible:
Byte 0 1 2 3 4 5 6 7
←
00_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H
Acus. pet. | Estado | irrelevante
Estado:
0000_H = Ningún mensaje recibido disponible.
0001_H = Mensaje recibido disponible.
$0B01_H$ = El búfer de recepción está lleno en más de 2/3 partes.</td><td></td></tr>
<tr><td></td><td>La CPU escribe petición para recibir:
Byte 0 1 2 3 4 5 6 7
20_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H →
Petición | Irrelevante</td><td></td></tr>
<tr><td>Ciclo siguiente (n+1)</td><td>El programa de usuario lee la respuesta siguiente del módulo (el módulo acusa la recepción, responde con el primer segmento y aumenta el número secuencial):
←
21_H | 0006_H | ‘a’ | ‘b’ | ‘c’ | ‘d’ | ‘e’
Acus. pet. | Longitud | Datos</td><td></td></tr>
<tr><td></td><td>La CPU escribe la petición para acusar el 1er segmento:
21_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H →
Petición | irrelevante</td><td></td></tr>
<tr><td>Ciclo siguiente (n + 2)</td><td>El programa de usuario lee el 2º segmento del módulo:
←
22_H | ‘f’ | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H
Acus. pet. | Datos | irrelevante</td><td></td></tr>
</table>

<table>
<tr><th>Ciclo de la CPU</th><th>CPU escribe en ET 200S 1SI</th><th>CPU lee del ET 200S 1SI</th></tr>
<tr><td></td><td colspan="2">La CPU escribe la petición para acusar el 2º segmento:
<table><tr><td>22_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Petición</td><td colspan="7">irrelevante</td></tr></table>→</td></tr>
<tr><td>Ciclo siguiente (n + 3)</td><td colspan="2">El módulo regresa al estado de reposo después de finalizar la primera transacción de recepción.
←<table><tr><td>00_H</td><td>$nnnn_H$</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Acus. pet.</td><td>EstadoUS</td><td colspan="5">irrelevante</td></tr></table></td></tr>
<tr><td>—</td><td colspan="2">La CPU finaliza la petición.</td></tr>
</table>

2.12.4 Ejemplo de proceso al leer el estado de señalización V.24

Ejemplo de proceso

La tabla siguiente muestra un ejemplo de cómo la CPU lee el estado de las señales V.24 desde el módulo interfaz serie. La memoria E/S es de 8 bytes.

Tabla 2- 25 Ejemplo de proceso al leer el estado de señal V.24

Ciclo de la CPU	CPU escribe en ET 200S 1SI	CPU lee del ET 200S 1SI
1.	El programa de usuario lee el código de reposo del módulo: Byte 0 1 2 3 4 5 6 7 ← 00_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H Acus. pet. | Estado | irrelevante	
	La CPU escribe la petición para leer el estado de señal V.24: Byte 0 1 2 3 4 5 6 7 30_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H → Petición | Irrilevante	
2.	El programa de usuario lee la respuesta siguiente del módulo: ← 31_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H Acus. pet. | Señales | irrelevante MSB LSB 00 | 0 | 0 | 0 | DCD | CTS | RTS | DSR | DTR 7 6 5 4 3 2 1 0	
	La CPU escribe el acuse y acepta el número secuencial. 31_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H → Petición | irrelevante	
3.	El módulo regresa al estado de reposo después de finalizar la primera transacción. ← 00_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H Acus. pet. | Estado | irrelevante	
—	La CPU finaliza la petición.	

2.12.5 Ejemplo de proceso al escribir señales V.24

Ejemplo de proceso al escribir señales V.24

La tabla siguiente muestra un ejemplo de cómo la CPU escribe las señales V.24 en el módulo interfaz serie. La memoria E/S es de 8 bytes.

Tabla 2- 26 Ejemplo de proceso al escribir señales V.24

Ciclo de la CPU	CPU escribe en ET 200S 1SI	CPU lee del ET 200S 1SI
1.	El programa de usuario lee el código de reposo del módulo: Byte 0 1 2 3 4 5 6 7 ← 00_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H Acus. pet. | Estado | irrelevante	
	La CPU escribe la petición para escribir las señales V.24: Byte 0 1 2 3 4 5 6 7 40_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H → Petición | Estados de señalización | irrelevante MSB LSB 00 | 0 | 0 | 0 | DCD | CTS | RTS | DSR | DTR 7 6 5 4 3 2 1 0	
2.	El programa de usuario lee la respuesta siguiente del módulo: ← 40_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H Acus. pet. | Estado | irrelevante	
	La CPU escribe el estado de reposo en el byte de petición: 00_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H | xx_H → Petición | irrelevante	
3.	El programa de usuario lee la siguiente respuesta del módulo (éste regresa al estado de reposo al finalizar la transacción): ← 00_H | $nnnn_H$ | xx_H | xx_H | xx_H | xx_H | xx_H Acus. pet. | Estado | irrelevante	
—	La CPU escribe el código de reposo en la petición y la finaliza.	

2.12.6 Parámetros para el control del flujo de datos

Parámetros para el control del flujo de datos

El código de petición para la transferencia de parámetros con el driver ASCII permite configurar más parámetros.
Esto depende del tipo de control del flujo de datos seleccionado en el archivo GSD. Los tres tipos de control del flujo de datos se describen en la tabla siguiente.

Tabla 2- 27 Parámetros para el control del flujo de datos

Trama de parámetros para el control del flujo de datos con XON/XOFF			
Byte	**Descripción**	**Rango de valores**	**Valor por defecto**
1	Número de bloque de parámetros	20_H	
2 y 3	Longitud	0004_H	0004_H
4	Carácter XON	0 a 127 (7 bits de datos) 0 a 255 (8 bits de datos)	11 (DC1)
5	Carácter XOFF	0 a 127 (7 bits de datos) 0 a 255 (8 bits de datos)	13 (DC3)
6 y 7	Tiempo de espera para XON tras XOFF	20 a 655350 en una escala de 10 ms	200 (2000 ms)
Trama de parámetros para el control del flujo de datos con RTS/CTS			
Byte	**Descripción**	**Rango de valores**	**Valor por defecto**
1	Número de bloque de parámetros	21_H	
2 y 3	Longitud	0002_H	0002_H
4 y 5	Tiempo de espera para CTS = ON	20 a 655350 en una escala de 10-ms	200 (2000-ms)
Trama de parámetros para el manejo automático de las señales cualificadoras RS232C			
Byte	**Descripción**	**Rango de valores**	**Valor por defecto**
1	Número de bloque de parámetros	22_H	
2 y 3	Longitud	0004_H	0004_H
4 y 5	Tiempo para RTS = OFF tras la transferencia	0 a 655350 en una escala de 10-ms	1 (10-ms)
6 y 7	Tiempo de espera para CTS = ON tras RTS = ON	0 a 655350 en una escala de 10-ms	1 (10-ms)

Ejemplo de procesos para XON/XOFF

La tabla siguiente muestra un ejemplo de cómo la CPU configura los parámetros XON/XOFF. La memoria E/S es de 4 bytes.

Tabla 2- 28 Ejemplo de procesos para XON/XOFF

<table>
<tr><th>Ciclo de la CPU</th><th>CPU escribe en ET 200S 1SI</th><th>CPU lee del ET 200S 1SI</th></tr>
<tr><td rowspan="2">1.</td><td>El programa de usuario ve el siguiente código de reposo del módulo: →</td><td>Byte 0 1 2 3
<table><tr><td>00_H</td><td>$nnnn_H$</td><td>xx_H</td></tr><tr><td>Acus. pet.</td><td>Estado</td><td>irrelev.</td></tr></table></td></tr>
<tr><td>Byte 0 1 2 3
<table><tr><td>50_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Petición</td><td colspan="3">irrelevante</td></tr></table></td><td>← Petición: Enviar código de parámetros (1 0 1 ó 5_H) más el número secuencial 0</td></tr>
<tr><td rowspan="4">2.</td><td>El programa de usuario lee la respuesta siguiente del módulo: →</td><td><table><tr><td>50_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Acus. pet.</td><td colspan="3">irrelevante</td></tr></table></td></tr>
<tr><td colspan="2">La CPU envía el 1er segmento, ya que la petición se ha aceptado.</td></tr>
<tr><td><table><tr><td>51_H</td><td>20_H</td><td>0004_H</td></tr><tr><td>Petición</td><td>Flujo de datos</td><td>Longitud de envío</td></tr></table></td><td>← Petición: Reanudar parámetros y aumentar número secuencial
← Flujo de datos: Código de parámetros de flujo de datos</td></tr>
<tr><td></td><td></td></tr>
<tr><td rowspan="4">3.</td><td>El programa de usuario lee la respuesta siguiente del módulo: →</td><td><table><tr><td>51_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Acus. pet.</td><td colspan="3">irrelevante</td></tr></table></td></tr>
<tr><td colspan="2">La CPU envía el 2º segmento, ya que no se indicó ningún error:</td></tr>
<tr><td><table><tr><td>52_H</td><td>$0B_H$</td><td>$0D_H$</td><td>00_H</td></tr><tr><td>Petición</td><td>DC1</td><td>DC3</td><td>Tiempo de espera para XON tras XOFF - msB</td></tr></table></td><td></td></tr>
<tr><td></td><td></td></tr>
<tr><td rowspan="4">4.</td><td>El programa de usuario lee la respuesta siguiente del módulo: →</td><td><table><tr><td>52_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Acus. pet.</td><td colspan="3">irrelevante</td></tr></table></td></tr>
<tr><td colspan="2">La CPU envía el 3er segmento, ya que no se indicó ningún error:</td></tr>
<tr><td><table><tr><td>53_H</td><td>$C8_H$</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Petición</td><td>Tiempo de espera para XON tras XOFF -LSB</td><td colspan="2">Irrelevant</td></tr></table></td><td></td></tr>
<tr><td></td><td></td></tr>
</table>

<table>
<tr><th>Ciclo de la CPU</th><th>CPU escribe en ET 200S 1SI</th><th>CPU lee del ET 200S 1SI</th></tr>
<tr><td rowspan="4">5.</td><td>El programa de usuario lee la respuesta siguiente del módulo: →</td><td><table><tr><td>53_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Acus.pet.</td><td colspan="3">irrelevante</td></tr></table></td></tr>
<tr><td colspan="2">La CPU repite el 3er segmento y espera el acuse de fin de petición.</td></tr>
<tr><td><table><tr><td>53_H</td><td>$C8_H$</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Petición</td><td>Tiempo de espera para XON tras XOFF - LSB</td><td colspan="2">irrelevante</td></tr></table></td><td></td></tr>
<tr><td rowspan="4">6.</td><td>El programa de usuario lee la respuesta siguiente del módulo: →</td><td><table><tr><td>73_H</td><td>$nnnn_H$</td><td>xx_H</td></tr><tr><td>Acus.pet.</td><td>Estado</td><td>irrelev.</td></tr></table></td></tr>
<tr><td colspan="2">La CPU escribe el código de reposo en la petición y la finaliza.</td></tr>
<tr><td><table><tr><td>00_H</td><td>xx_H</td><td>xx_H</td><td>xx_H</td></tr><tr><td>Petición</td><td colspan="3">irrelevante</td></tr></table></td><td></td></tr>
</table>

2.12.7 Tratamiento de errores

Condiciones de error

El módulo interfaz serie emite un error como reacción a las condiciones siguientes:

- Si la petición de envío es superior a 224 bytes, el módulo responde con un acuse de fin de petición y la palabra de estado contiene el código de error. A continuación, la CPU escribe un código de reposo en la petición y la finaliza.
- Si se ha enviado una petición de recepción al módulo y el mensaje recibido contiene un error, el módulo aplica el código de petición de recepción con el número secuencial cero y la palabra de estado contiene el código de error. A continuación, la CPU escribe un código de reposo en la petición y la finaliza.
- Si se ha enviado una petición de recepción al módulo y no hay ningún mensaje recibido disponible, el módulo aplica el código de petición de recepción con el número secuencial cero y la palabra de estado contiene el valor 0101_H. No es ninguna condición de error, pero impide que el módulo se bloquee en el modo de petición de recepción y que espere un mensaje recibido, de modo que puedan ejecutarse peticiones de envío. La CPU escribe un código de reposo en la petición y la finaliza.

Excepciones

Como ya se ha dicho, una operación determinada (p. ej. una petición de envío) no puede iniciarse en el programa de usuario antes de que el módulo esté en estado de reposo. Después de enviar una petición, la operación tiene que esperar a que el módulo acepte el código de la petición antes de ejecutar una operación determinada. En las operaciones con segmentación en el proceso pueden aparecer las excepciones siguientes:

Nota

En las descripciones siguientes de una operación para enviar o parametrizar, el emisor es la CPU y el receptor el módulo interfaz serie. En una operación para recibir, el emisor es el módulo interfaz serie y el receptor la CPU.

- **Error:** El emisor supervisa el bit de error del receptor en relación a una transacción segmentada. Con el bit de error activado, ocurre lo siguiente:
  - **La CPU es emisora** (petición de envío): La CPU vuelve a enviar los segmentos, comenzando por el segmento siguiente al número indicado por el receptor.
  - **El módulo es emisor** (petición de recepción): El módulo 1SI interrumpe la transmisión del telegrama Rx al usuario con el mensaje de error 0x0551 en la palabra de estado. El módulo espera al acuse del mensaje de error. Una vez finalizada la secuencia de recepción en curso, el telegrama Rx interrumpido se vuelve a notificar al usuario o bien se pone a disposición.
- **El número secuencial no está en el orden correcto:**Cuando, durante una operación segmentada, el receptor recibe un segmento con un número secuencial que no es el número secuencial anterior + 1, debe comunicar en la respuesta un error y el último número secuencial recibido.
  - **La CPU es receptora** (petición de recepción): Cuando la CPU recibe un segmento con bit de estado activado y mensaje de error 0x0551 en la palabra de estado, debe interrumpir la petición de recepción y descartar los datos aplicados anteriormente.
- **Código de petición modificado:**
  - Si el receptor recibe un segmento con un código de petición distinto al código con el que se empezó la operación segmentada y que no sea ni 000 ni 111, el receptor ignora el otro código y desecha los datos correspondientes.
  - Si el receptor recibe un segmento con el código de petición del estado de reposo durante una operación segmentada, la operación se cancela y se adopta el estado de reposo sin que se active ningún bit de error.
  - Si el receptor recibe un segmento con el código de petición del acuse de fin de petición durante una operación segmentada, la operación se cancela y se adopta el estado de reposo sin que se active ningún bit de error.
  - Si durante una operación segmentada el emisor recibe una respuesta que tiene un código de petición distinto, el mensaje tiene que cancelarse. A continuación se envía de nuevo el código de reposo, el módulo tiene que pasar al estado de reposo y ejecutar de nuevo la operación.

2.13 Diagnóstico

Resumen

Las funciones de diagnóstico del módulo ET 200S 1SI permiten localizar rápidamente los errores que puedan haber surgido durante el funcionamiento. Dispone de las siguientes posibilidades de diagnóstico:

- Diagnóstico a través de los LEDs de estado en el panel frontal del módulo ET 200S 1SI
- Diagnóstico a través de la salida STATUS de los bloques de función
- Diagnóstico a través del diagnóstico de esclavos PROFIBUS

Información de diagnóstico a través de LEDs de estado

Los siguientes LEDs de estado se encuentran en el panel frontal del módulo interfaz ET 200S 1SI:

- **TX** (verde): Se enciende cuando el módulo envía datos a través de la interfaz.
- **RX** (verde): Se enciende cuando el módulo recibe datos a través de la interfaz.
- **SF** (rojo): Indica un posible error:
  - Error de hardware
  - Error de parametrización
  - Rotura de hilo o cables sueltos entre el módulo y el interlocutor:

    Se detecta únicamente en conexiones de interfaz RS-422 con el parámetro Preasignación de la línea de recepción = R(A) 5V/R(B) 0V.
  - Error de comunicación (paridad, error de trama, desbordamiento del búfer)

Estructura de los mensajes de diagnóstico de los bloques de función

Para el diagnóstico de errores, cada bloque de función dispone de un parámetro STATUS. Cada número de mensaje STATUS tiene el mismo significado, independientemente del bloque de función utilizado. La figura siguiente muestra la estructura del parámetro STATUS.

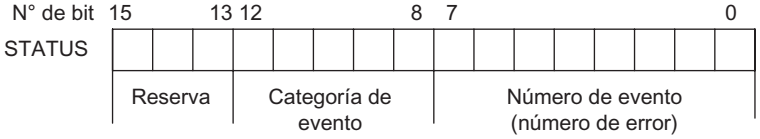


Figura 2-23 Estructura del parámetro STATUS

Ejemplo: La figura siguiente muestra el contenido del parámetro STATUS para el evento "Interrupción de petición por rearranque completo, rearranque o reset" (clase de evento $1E_H$, número de evento $0D_H$).

Evento: “Cancelación de petición a causa de rearranque completo, rearranque o reinicialización”

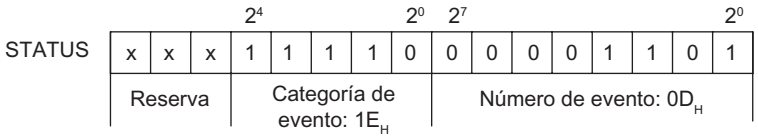


Figura 2-24 Ejemplo: Parámetro STATUS para la clase de evento 1EH, evento 0DH

Mensajes de diagnóstico de los bloques de función

La tabla siguiente describe las clases de evento, las definiciones de los números de evento y el remedio recomendado para cada condición de error.

Tabla 2- 29 Mensajes de diagnóstico en el parámetro STATUS

Número de evento	Evento	Solución
Clase de evento 2 ($0x02_H$): "Error al inicializar"		
(02) 01_H	Sin parametrización (válida).	Asigne parámetros correctos al módulo. Dado el caso, compruebe que el equipo se haya instalado de forma adecuada.
Clase de evento 5 (05_H): "Error al procesar una petición de la CPU"		
(05) 02_H	La petición no está permitida en este estado operativo del módulo ET 200S 1SI (p. ej. interfaz del equipo no parametrizada).	El telegrama de envío tiene más de 224 bytes. La petición de envío ha sido cancelada por el módulo ET 200S 1SI. Seleccione una longitud de telegrama menor.
(05) $0E_H$	Longitud de telegrama no válida	El telegrama de envío tiene más de 224 bytes. La petición de envío ha sido cancelada por el módulo ET 200S 1SI. Seleccione una longitud de telegrama menor.
(05) 50_H	Petición de actualización de parámetros no válida para el control del flujo de datos actual del módulo ET 200S 1SI.	Modifique los parámetros del bloque de función (FB6 S_XON, FB7 S_RTS, FB8 S_V24) en el programa del autómata o modifique el control del flujo de datos del módulo ET 200S 1SI en la configuración de hardware, de modo que ambos coincidan.
(05) 51_H	Error de secuencia de trama en la comunicación entre el módulo ET 200S 1SI y el sistema de automatización. El error ha surgido al transmitir un telegrama recibido del módulo ET 200S 1SI en el sistema de automatización.	El módulo y el sistema de automatización han cancelado la transferencia. Repita la petición de recepción; el módulo ET 200S 1SI vuelve a enviar el mensaje recibido.

Número de evento	Evento	Solución
Clase de evento 7 (07H): "Error de envío"		
(07) 02_H	Sólo con 3964(R): Error al establecer la conexión: Tras el envío de STX se ha recibido NAK o un carácter cualquiera (excepto DLE o STX).	Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(07) 03_H	Sólo con 3964(R): Tiempo de retardo de acuse (TRA) transcurrido: Tras en envío de STX no ha llegado ninguna respuesta del interlocutor dentro del tiempo de retardo de acuse.	El dispositivo interlocutor es demasiado lento o no está preparado para recibir o bien hay p. ej. una rotura de la línea de envío. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(07) 04_H	Sólo con 3964(R): Interrupción por parte del interlocutor: Durante el proceso de envío en curso el interlocutor ha recibido uno o varios caracteres.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(07) 05_H	Sólo con 3964(R): Acuse negativo al enviar	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(07) 06_H	Sólo con 3964(R): Error al terminar la conexión: • El telegrama ha sido rechazado por el interlocutor con NAK u otro carácter al final (con la excepción de DLE) o bien • El carácter de confirmación (DLE) se ha recibido demasiado pronto.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(07) 07_H	Sólo con 3964(R): Tiempo de retardo de acuse al terminar la conexión/tiempo de supervisión de respuesta tras el telegrama de envío agotados: Tras la interrupción de la conexión con DLE ETX no ha llegado ninguna respuesta del interlocutor dentro del TRA.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe si es así con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
(07) 08_H	Sólo con driver ASCII: El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor tiene un defecto, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.

Número de evento	Evento	Solución
(07) $0B_H$	Sólo con 3964(R): El conflicto de inicialización no puede resolverse, puesto que ambos interlocutores tienen predeterminada prioridad alta.	Cambie la parametrización.
(07) $0C_H$	Sólo con 3964(R): El conflicto de inicialización no puede resolverse, puesto que ambos interlocutores tienen ajustada prioridad baja.	Cambie la parametrización.
Clase de evento 8 (08_H): "Error de recepción"		
(08) 02_H	Sólo con 3964(R): Error al establecer la conexión: • En posición de reposo se han recibido uno o más caracteres (cualquiera con la excepción de NAK o STX) o bien • Tras la recepción de un STX el interlocutor ha enviado otros caracteres sin esperar la recepción de la respuesta DLE. Tras Power ON del interlocutor: • Mientras el interlocutor se conecta, el módulo recibe un carácter indefinido.	Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(08) 05_H	Sólo con 3964(R): Error lógico durante la recepción: Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado del telegrama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(08) 06_H	Tiempo de retardo entre caracteres (TRC) transcurrido: • No se han recibido dos caracteres consecutivos dentro del TRC o bien Sólo con 3964(R): • 1. El 1er carácter tras el envío de DLE en el establecimiento de la conexión no se ha recibido dentro del TRC.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(08) 07_H	Sólo con 3964(R): Longitud de telegrama no permitida: Se ha recibido un telegrama de longitud 0.	La recepción de un telegrama de longitud 0 no es un error. Compruebe el motivo por el que el interlocutor envía telegramas sin datos útiles.

Número de evento	Evento	Solución
(08) 08_H	Sólo con 3964(R): Error en el carácter de comprobación de bloque BCC: El valor del BCC determinado internamente no concuerda con el BCC recibido por el interlocutor al final de la conexión.	Compruebe si el enlace tiene grandes interferencias; en ese caso, ocasionalmente también pueden aparecer códigos de error. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
(08) 09_H	Sólo con 3964(R): El número de repeticiones debe ajustarse igual.	Parametrice en el interlocutor el mismo tiempo de espera de bloque que en el módulo. En caso necesario, compruebe el comportamiento erróneo del interlocutor con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
(08) $0A_H$	No hay un búfer de recepción libre disponible: Durante la recepción no se disponía de ningún búfer de recepción vacío.	El FB S_RCV debe ser llamado con mayor frecuencia.
Clase de evento 8 (08_H): "Error de recepción"		
(08) $0C_H$	Error de transmisión: • Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de rebosamiento). Sólo con 3964(R): • Si esto ocurre durante el modo de envío o recepción, se inician repeticiones. • Si durante la posición de reposo se recibe un carácter defectuoso, el error se indica inmediatamente para facilitar la detección prematura de fallos en la línea de transmisión. • Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión de ambos interlocutores.	Los fallos en la línea de transmisión provocan repeticiones del telegrama reduciendo el paso de datos útiles. El riesgo de que se produzca un error no detectado aumenta. Modifique la configuración de su sistema o la disposición de la línea. Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos.
(08) $0D_H$	BREAK: La línea de recepción que llega al interlocutor está interrumpida.	Restablezca la conexión o conecte el interlocutor.
(08) 10_H	Sólo con driver ASCII: Error de paridad: • Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.
(08) 11_H	Sólo con driver ASCII: Error de trama de caracteres: • Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.

Número de evento	Evento	Solución
(08) 12_H	Sólo con driver ASCII: Después de que el módulo enviara XOFF o pusiera CTS a OFF, se recibieron más caracteres.	Vuelva a parametrizar el interlocutor o lea los datos del módulo con más rapidez.
(08) 18_H	Sólo con driver ASCII: DSR = OFF o CTS = OFF	Antes o durante un proceso de envío el interlocutor ha puesto todas las señales DSR o CTS a "OFF". Revise el control de las señales cualificadoras RS 232C en el interlocutor.
(08) 50_H	La longitud del telegrama de recepción es superior a 224 bytes o mayor que la longitud de telegrama parametrizada	Adaptar la longitud de telegrama del interlocutor
Clase de evento 11 ($0B_H$): Advertencia		
(0B) 01_H	El búfer de recepción está lleno en más de 2/3 partes	
Clase de evento 30 ($1E_H$): "Error en la comunicación entre el módulo y la CPU"		
(1E) $0D_H$	"Interrupción del trabajo por rearranque completo, rearranque o reset"	
(1E) $0E_H$	Error estático al llamar la SFC DPRD_DAT. Se puede disponer del valor de retorno RET_VAL de la SFC en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
(1E) $0F_H$	Error estático al llamar la SFC DPWR_DAT. Se puede disponer del valor de retorno RET_VAL de la SFC en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
(1E) 10_H	Error estático al llamar la SFC RD_LGADR. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
(1E) 11_H	Error estático al llamar la SFC RDSYSST. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
(1E) 20_H	Parámetro fuera del rango.	Cambie la entrada del bloque de función para que esté en el área válida.
(1E) 41_H	El número de bytes indicado en el parámetro LEN de los FB no está permitido	Respete el rango de 1 a 224 bytes.

Evaluación de la variable SFCERR

Obtendrá más información sobre los errores ocurridos (1E) $0E_H$, (1E) $0F_H$, (1E) 10_H y (1E) 11_H de la clase de evento 30 a través de la variable SFCERR.

La variable SFCERR se puede cargar desde el DB de instancia del correspondiente bloque de función.

Los mensajes de error registrados en la variable SFCERR se encuentran en las funciones de sistema "DPRD_DAT" y SFC15 "DPWR_DAT" en el manual de referencia *Software de sistema para S7-300 y S7-400. Funciones estándar y funciones de sistema*.

Diagnóstico de esclavos PROFIBUS

El diagnóstico de esclavos se atiene a la norma EN 50170, volumen 2, PROFIBUS. Dependiendo del maestro DP, el diagnóstico puede leerse con STEP 5 o STEP 7 para todos los esclavos DP que cumplan con la norma mencionada.

El diagnóstico de esclavos PROFIBUS abarca el diagnóstico y estado de módulos y el diagnóstico de canal. Encontrará información detallada sobre el diagnóstico de esclavos DP en el manual *Sistema de periferia descentralizada ET 200S*.

Diagnóstico de canal: El diagnóstico de canal proporciona información sobre errores de canal de módulos y comienza tras el estado del módulo. La tabla siguiente lista los tipos de error de canal.

Tabla 2- 30 Tipos de error de canal en el módulo interfaz serie ET 200S 1SI

Tipo de error	Significado	Solución
00110: Rotura de hilo	Cable roto o suelto.	Compruebe el cableado de los bornes. Compruebe el cable del interlocutor.
00111: Rebase por exceso	Desbordamiento del búfer; desbordamiento de la longitud del mensaje	El FB S_RCV debe ser llamado con mayor frecuencia.
01000: Rebase por defecto	Sólo con 3964(R): Se ha enviado un mensaje con longitud 0.	Compruebe el motivo por el que el interlocutor envía telegramas sin datos útiles.
01001: Fallo	Ha aparecido un error interno del módulo.	Cambie el módulo.
10000: Error de parametrización	Módulo no parametrizado.	Corrija la parametrización.
10110: Error de mensaje	Error de trama, error de paridad	Compruebe los ajustes de comunicación.

2.14 Datos técnicos

Datos técnicos generales

Para el módulo interfaz serie ET 200S 1SI 3964/ASCII rigen los datos técnicos generales, tal y como se recogen en el capítulo "Especificaciones técnicas generales" del manual *Sistema de periferia descentralizado ET 200S*. Encontrará dicho manual en:

http://www.siemens.com/simatic-tech-doku-portal

Especificaciones técnicas de los protocolos y de la interfaz

Tabla 2- 31 Datos técnicos generales del módulo ET 200S 1SI

Datos técnicos generales	
Elementos de indicación	• LED (verde): TX (enviar) • LED (verde): RX (recibir) • LED (rojo): SF (error colectivo)
Drivers de protocolo suministrados	Driver 3964(R) Driver ASCII
Velocidades de transferencia con el protocolo 3964(R) Velocidades de transferencia con drivers ASCII	110, 300, 600, 1200, 2400, 4800, 9600, 19.200, 38.400, 57.600, 76.800, 115.200
Trama de caracteres (10 bits u 11 bits)	Cantidad de bits por carácter: 7 u 8 Número de bits de arranque/parada: 1 ó 2 Paridad: Sin, par, impar, cualquiera
Memoria necesaria para los bloques estándar (FB)	Enviar y recibir: aprox. 4300 bytes
Datos técnicos de la interfaz RS 232C	
Interfaz	RS232C, 8 bornes
Señales RS 232C	TXD, RXD, RTS, CTS, DTR, DSR, DCD, PE Todas aisladas eléctricamente de la fuente de alimentación interna del módulo ET 200S 1SI.
Recorrido máximo de transferencia	15 m

Datos técnicos generales	
Datos técnicos de la interfaz RS-422/485	
Interfaz	• RS-422, 5 bornes • RS-485, 3 bornes
Señales RS-422 Señales RS-485	TXD (A)-, RXD (A)-, TXD (B)+, RXD (B)+, PE R/T (A)-, R/T (B)+, PE Todas aisladas eléctricamente de la fuente de alimentación interna del módulo ET 200S 1SI.
Recorrido máximo de transferencia	1200 m

Datos técnicos

Dimensiones y peso	
Dimensiones A x A x P (en mm)	15 x 81 x 52
Peso	aprox. 50 g
Datos específicos del módulo	
RS-232C • Número de entradas • Número de salidas	 4 3
RS-422 • Número de pares de entradas • Número de pares de salidas	 1 1
RS-485 Número de pares E/S	 1
Longitud de cable • Apantallado (RS232C) • Apantallado (RS-422/485)	 máx. 15 m máx. 1200 m
Grado de protección[1]	IEC 801-5
Tensiones, intensidades, potenciales	
Tensión nominal de alimentación de la electrónica (L +) • Protección contra cambio de polaridad	24V DC Sí
Aislamiento galvánico • Entre los canales y el bus posterior • Entre los canales y la tensión de alimentación de la electrónica • Entre los canales • Entre los canales y PROFIBUSDP	 Sí Sí No Sí

Dimensiones y peso	
Aislamiento ensayado con	
• Canales respecto al bus posterior y la tensión de carga L+	DC 500 V
• Tensión de carga L+ respecto al bus posterior	500 V AC
Fuente de corriente	
• Del bus posterior	máx. 10 mA
• De la fuente de alimentación L+	máx. 120 mA, típ. 50 mA
Disipación del módulo	típ. 1,2 W
Estado, alarmas, diagnóstico	
Indicador de estado	• LED verde (TX) • LED verde (RX)
Funciones de diagnóstico	
• Indicador de error colectivo	LED rojo (SF)
• Puede visualizarse información de diagnóstico	Posible
Salidas	
Salida, área RS232C	± máx. 10 V
• Con carga capacitiva	máx. 2500 pF
• Protección contra cortocircuito	Sí
• Corriente de cortocircuito	aprox. 60 mA
• Tensión en las salidas o entradas a PE (tierra)	máx. 25 V
Salida, RS-422/485	
• Resistencia de carga	mín. 50 kΩ
• Protección contra cortocircuito	Sí
• Corriente de cortocircuito	aprox. 60 mA
[1] Equipos de protección externos necesarios en las líneas de entradas de la tensión de usuario: • Blitzductor adaptador estándar del perfil soporte • Blitzductor tipo de módulo de protección KT AD24V	

Modbus/USS

3

3.1 Presentación del producto

Número de referencia

6ES7 138-4DF11-0AB0

Descripción del producto

El módulo interfaz serie ET 200S Modbus/USS es un módulo enchufable de la serie de productos ET 200S, que permite acceder a la comunicación en serie con la ayuda de tres interfaces de hardware (RS232C, RS422 y RS485) y dos protocolos de software:

- Modbus
- Maestro USS

Con el módulo interfaz serie ET 200S Modbus/USS es posible intercambiar datos entre sistemas de automatización u ordenadores a través de un acoplamiento punto a punto. Toda la comunicación se realiza mediante transferencias asíncronas en serie.

El tipo de comunicación se elige al parametrizar el módulo en la configuración de hardware de STEP 7 o en otra aplicación de configuración. En el catálogo de hardware aparecen los nueve modelos siguientes del módulo:

- Maestro Modbus (4 bytes)
- Maestro Modbus (8 bytes)
- Maestro Modbus (32 bytes)
- Esclavo Modbus (4 bytes)
- Esclavo Modbus (8 bytes)
- Esclavo Modbus (32 bytes)
- Maestro USS (4 bytes)
- Maestro USS (8 bytes)
- Maestro USS (32 bytes)

La transferencia de datos de 8 ó 32 bytes aumenta la eficacia de caudal, pero requiere más espacio E/S en el bastidor ET 200S. La transferencia de datos de 4 bytes no ocupa tanto espacio E/S en el bastidor ET 200S pero su eficacia de caudal es menor. La variante del módulo depende de los requisitos de cada aplicación.

Funciones del módulo interfaz serie ET 200S Modbus/USS

El módulo interfaz serie ET 200S Modbus/USS ofrece las funciones siguientes:

- Interfaz integrada según RS232C, RS-422 ó RS-485
- Velocidad de transmisión hasta 115,2 Kbaudios, semidúplex
- Integración de los siguientes protocolos de transmisión en el firmware del módulo:
  - Driver maestro Modbus
  - Driver esclavo Modbus
  - Driver maestro USS

La parametrización del módulo determina las funciones de los drivers.

La tabla siguiente lista las funciones de las diferentes interfaces del driver.

Tabla 3- 1 Funciones de los drivers del módulo interfaz serie ET 200S Modbus/USS

Función	RS-232C	RS-422	RS-485
Driver Modbus	Sí	Sí	Sí
Manejo automático de las señales RS232C	Sí	No	No
Driver maestro USS	Sí	No	Sí

Nota

¡Detrás de los CPs de comunicación externos CP342-5 (Profibus DP) y CP343-1 (Profinet IO) no puede emplearse el módulo ET 200S Modbus/USS con los FBs estándar normales!

Para el uso del módulo después del CP de comunicaciones CP 342-5 (Profibus DP) o CP 343-1 (Profinet IO) existen FBs especiales disponibles en las páginas de Customer Support:

Véase http://support.automation.siemens.com/WW/view/com/26263724

Indicadores LED

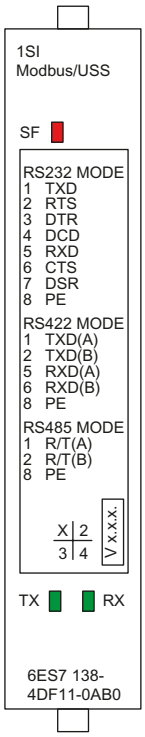
Los siguientes indicadores LED se encuentran en el panel frontal del módulo interfaz:

LED	Color	Descripción
SF	Rojo	Indicador de error colectivo
TX	Verde	La interfaz está enviando.
RX	Verde	La interfaz está recibiendo.

Los estados operativos y los errores indicados por estos LEDs se describen en el apartado Información de diagnóstico de los LEDs de estado (Página 227).

Panel frontal

La figura muestra la rotulación del panel frontal del módulo interfaz serie ET 200S Modbus/USS.



3.2 Instrucciones breves sobre la puesta en marcha del módulo interfaz serie

Tarea

En esta breve descripción se muestra un ejemplo sobre el envío y la recepción de datos entre módulos interfaz serie, que sirve de base para explicar cómo se crea una aplicación operativa, cómo funcionan las operaciones básicas del módulo interfaz serie (hardware y software) y cómo se verifican el hardware y el software.

En este ejemplo se utilizan dos módulos interfaz serie ET 200S 1SI Modbus/USS como acoplamiento RS232C maestro Modbus <-> esclavo Modbus.

Requisitos

Deben cumplirse los requisitos siguientes

- Se pone en marcha una estación ET 200S en un equipo S7 con maestro DP.
- Se necesitan los componentes siguientes:
  - Dos módulos terminal TM-E15S24-01
  - Dos módulos interfaz serie ET 200S 1SI Modbus/USS
  - El material de cableado necesario

Montar, cablear y equipar

Monte y efectúe el cableado de ambos módulos de terminal TM-E15S24-01 (véase la siguiente figura). Conecte los dos módulos interfaz serie ET 200S 1SI Modbus/USS con los módulos de terminal. (Encontrará información detallada en el manual *Periferia descentralizada*).

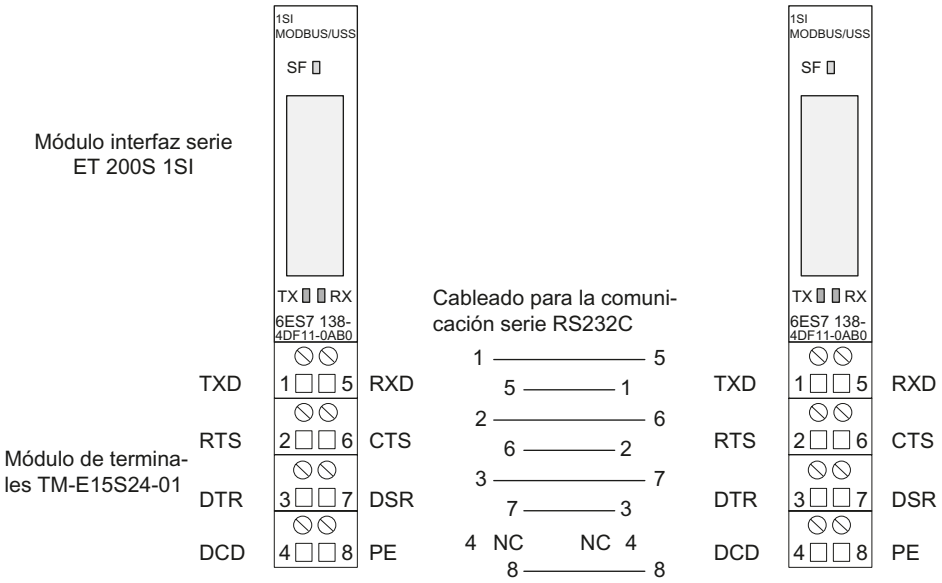


Figura 3-1 Asignación de pines del ejemplo

Configuración utilizada

En la siguiente tabla encontrará la configuración utilizada para el programa de ejemplo.

Tabla 3- 2 Parametrización de la aplicación de ejemplo

Parámetro	Valor
Diagnóstico colectivo	Bloquear
Interfaz	RS232C
Línea de recepción preasignada	
Modo de operación	Funcionamiento normal
Dirección de esclavo [1]	1
Control del flujo de datos (preajuste)	Sin
Velocidad de transmisión	9600
Bits de parada	1
Paridad	Par
Múltiplo del tiempo de ejecución	1
Tiempo de respuesta (ms) [2]	2000

Parámetro	Valor
Tiempo para RTS = off (ms)	
Tiempo de espera para la valoración de los datos (ms)	
Borrar el búfer de recepción durante el arranque	Sí
[1] sólo para esclavo Modbus [2] sólo para maestro Modbus	

Bloques utilizados

En la siguiente tabla encontrará los bloques utilizados en el programa de ejemplo.

Bloques	Símbolo	Comentario
OB 1	CYCLE	Ejecución cíclica del programa
OB 100	RESTART	Ejecución de rearranque completo
DB 21	SEND_IDB_SI_0	DB de instancia para FB S_SEND_SI
DB 22	RECV_IDB_SI_1	DB de instancia para FB S_RECV_SI
DB 40	SEND_WORK_DB_SI_0	DB de trabajo para FB estándar 3
DB 41	RECV_WORK_DB_SI_1	DB de trabajo para FB estándar 2
DB 42	SEND_SRC_DB_SI_0	Bloque de datos de envío
DB 43	RECV_DST_DB_SI_0	Bloque de datos de recepción
DB 81	MODSL_IDB_SI_1	DB de instancia para FB S_MODB
DB 100	CONVERSION_DB	DB de conversión para FB S_MODB
FB 2	S_RECV_SI	FB estándar para recibir datos
FB 3	S_SEND_SI	FB estándar para enviar datos
FB 81	S_MODB	FB estándar para comunicación esclavo Modbus
FC 10	Iniciación	Inicializar bloques de datos
FC 21	SEND_SI_0	Enviar datos
FC 22	RECV_SI_1	Recibir datos

Suministro e instalación

El programa de ejemplo del módulo ET 200S 1SI Modbus/USS junto con los bloques de función están disponibles en la dirección de Internet siguiente:

http://support.automation.siemens.com/WW/view/es/10805265/133100

Después de la instalación, el programa de ejemplo se encuentra en el proyecto zXX21_11_1SI_MODBUS.

El proyecto se abre en el SIMATIC Manager de STEP 7 con el comando de menú "Archivo > Abrir > Proyectos de ejemplo".

El ejemplo de programación está disponible de forma compilada y en forma de archivo fuente ASCII. Asimismo, hay una tabla de símbolos con los símbolos empleados en el ejemplo.

Si no se dispone de un segundo ET 200S 1SI Modbus/USS como interlocutor, hay que eliminar el segundo ET 200S 1SI Modbus/USS en HW Config con el comando "Edición > Borrar". Además es necesario eliminar en el OB 1 el comentario de la llamada del FB 81 (esclavo Modbus FB).

Cargar en la CPU

El hardware para el ejemplo está completamente montado y la unidad de programación está conectada.

Tras el borrado total de la CPU (estado operativo STOP), transfiera el ejemplo completo a la memoria de usuario. A continuación conmute el interruptor de modo de operación de STOP a RUN.

Comportamiento erróneo

Si durante el arranque se produce un error, los accesos cíclicos a los módulos no se llevan a cabo, y se activa la indicación de error.

Si se produce un mensaje de error, se activa el parámetro de salida ERROR de los bloques. En el parámetro STATUS de los bloques hay una descripción más detallada de los errores. Si el mensaje de error 16#1E0E o 16#1E0F se halla en STATUS, la descripción detallada del error se encuentra almacenada en las variables SFCERR del DB de instancia.

Conexión, programa de arranque

El programa de arranque se halla en el OB 100.

En el arranque se desactivan los bits de control y los contadores

Programa cíclico

El programa cíclico se halla en el OB 1.

En el ejemplo, los bloques de función FB 2 S_RECV_SI y FB 3 S_SEND_SI operan para el maestro Modbus conjuntamente con las funciones FC 21 y FC 22, así como con los bloques de datos DB 21 y DB 22 como DBs de instancia y con DB 42 y DB 43 como DB emisor o receptor.

Para el esclavo Modbus trabaja el FB 81 S_MODB con el DB 81 como DB de instancia y el DB 100 como DB de conversión.

La parametrización de los bloques de función de este ejemplo se produce en parte con constantes y en parte con operandos actuales direccionados simbólicamente.

Descripción

En la transmisión de datos, el ET 200S 1SI Modbus/USS enchufado en el slot 2 (maestro Modbus) "coge" datos del ET 200S 1SI Modbus/USS del slot 3 (esclavo Modbus). Si trabaja con otro interlocutor, desaparece la llamada del FB 81 (S_MODB).

Descripción de la FC 21 (SEND)

Sección del programa "Generate edge S_SEND_SI_REQ":

El S_SEND_SI se ejecuta una vez al comienzo con S_SEND_SI_ REQ=0. A continuación, S_SEND_SI_ REQ recibe el valor 1. Si en el parámetro de control S_SEND_SI_ REQ se detecta un cambio de estado de señal de 0 a 1, se inicia la petición S_SEND_SI.

Con S_SEND_SI_ DONE=1 ó S_SEND_SI_ERROR=1, S_SEND_SI_REQ vuelve a adquirir del valor 0.

Sección del programa "S_SEND_SI_DONE=1":

Si la transferencia finaliza con éxito, en la salida de parámetro de S_SEND_SI, el parámetro S_SEND_SI_DONE recibe el valor 1.

A fin de posibilitar la distinción de transferencias consecutivas, en la palabra de datos 18 del bloque de trabajo DB 40 se añade un contador de envíos S_SEND_SI_WORK_CNT_OK.

Sección del programa "S_SEND_SI_ERROR=1":

Si S_SEND_SI se ejecuta con S_SEND_SI_ERROR=1, en la palabra de datos 20 el contador de errores S_SEND_SI_WORK_CNT_ERR incrementa. Además, se hace una copia de S_SEND_SI_WORK_STAT, ya que en el próximo acceso será sobrescrito con 0 y ya no podrá consultarse.

Descripción de la FC 22 (RECEIVE)

Sección del programa "Enable Receive Data":

Para recibir datos es necesario que la habilitación de recepción S_RECV_SI_EN_R del bloque S_RECV_SI tenga el valor 1.

Sección del programa "S_RECV_SI_NDR=1":

Si S_RECV_SI_NDR está activado, significa que se han recibido datos nuevos y el contador de recepción S_RECV_SI_WORK_CNT_OK incrementa.

Sección del programa "S_RECV_SI_ ERROR=1":

En caso de error durante el arranque, es decir, si el bit de error de la salida del parámetro S_RECV_SI está activado, el contador S_RECV_SI_WORK_CNT_ERR se incrementa. Además, se hace una copia de S_RECV_SI_WORK_STAT, ya que en el próximo acceso será sobrescrito con 0 y ya no podrá consultarse.

Todos los valores relevantes pueden consultarse en la tabla de variables con fines de comprobación.

Descripción del DB 42

Con la petición Función Code 1 (Read Coil Status) configurada en el presente ejemplo se pretende que el esclavo Modbus que tiene la dirección "1" lea 16 bits a partir de la dirección inicial "0". Los 16 bits leídos se guardan con la FC 22 (RECV) en el DB de recepción (DB43) a partir de la dirección offset 0.

Los parámetros de la petición de maestro Modbus (FC 21 (SEND)) están registrados en el DB de envío (DB 24). Véase la tabla siguiente:

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUC		
+ 0.0	slave_adress	BYTE	B#16#01	Del esclavo Modbus "1"
+ 1.0	function_code	BYTE	B#16#01	Con FC 1 (Read Coil Status)
+ 2.0	bit_start_adr	WORD	W#16#0000	A partir de la dirección inicial Modbus 0
+ 4.0	bit_count	INT	16	Leer 16 bits (1 palabra)
+ 6.0	a	ARRAY [1…1194]		
* 1.0		BYTE		
= 1200.0		END_STRUCT		

Iniciar petición de maestro Modbus

¡Para iniciar la petición de maestro Modbus debe activar la marca M 120.7 TRUE en la VAT!

Descripción del DB 100

En el lado del esclavo Modbus, los datos solicitados se preparan mediante la llamada del FB 81 (S_MODB).

Las direcciones utilizadas en el telegrama del maestro Modbus se guardan de la siguiente manera en el área de datos SIMATIC en el DB de conversión configurado (DB 100):

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	FC01_MOD_STRT_ADR_1	WORD	W#16#0	Representación de las direcciones Modbus 0 a 255 en el área de marcas SIMATIC a partir de 0
+2.0	FC01_MOD_END_ADR_1	WORD	W#16#0FF	
+4.0	FC01_CNV_TO_FLAG_A	WORD	W#16#0	
+6.0	FC01_MOD_STRT_ADR_2	WORD	W#16#100	
+8.0	FC01_MOD_END_ADR_2	WORD	W#16#1FF	
+10.0	FC01_CNV_TO_OUTPUT	WORD	W#16#0	
+12.0	FC01_MOD_STRT_ADR_3	WORD	W#16#200	
+14.0	FC01_MOD_END_ADR_3	WORD	W#16#2FF	
+16.0	FC01_CNV_TO_TIMER	WORD	W#16#0	
+18.0	FC01_MOD_STRT_ADR_4	WORD	W#16#300	
+20.0	FC01_MOD_END_ADR_4	WORD	W#16#3FF	
+22.0	FC01_CNV_TO_COUNTER	WORD	W#16#0	
+24.0	FC02_MOD_STRT_ADR_5	WORD	W#16#0	
+26.0	FC02_MOD_END_ADR_5	WORD	W#16#0FF	
+28.0	FC02_CNV_TO_FLAG_B	WORD	W#16#0	
+30.0	FC02_MOD_STRT_ADR_6	WORD	W#16#100	
+32.0	FC02_MOD_END_ADR_6	WORD	W#16#2FF	
+34.0	FC02_CNV_TO_INPUT	WORD	W#16#0	
+36.0	FC03_06_16_DB_NO	WORD	W#16#02A	
+38.0	FC04_DB_NO	WORD	W#16#02A	
+40.0	DB_MIN	WORD	W#16#02A	
+42.0	DB_MAX	WORD	W#16#02A	
+44.0	FLAG_MIN	WORD	W#16#0	Área de marcas 0 a 255 habilitada
+46.0	FLAG_MAX	WORD	W#16#0FF	

Dirección	Nombre	Tipo	Valor inicial	Comentario
+48.0	OUTPUT_MIN	WORD	W#16#0	
+50.0	OUTPUT_MAX	WORD	W#16#0FF	
=52.0		END_STRUCT		

En el ejemplo concreto se representan las direcciones Modbus 0 a 255 solicitadas con una FC 1 a través de las direcciones 0 a 4 del DB 100 en el área de marcas SIMATIC a partir de 0.

Después de las direcciones del DB 100 44 y 46, el área de marcas SIMATIC 0 a 255 está habilitada para peticiones del maestro Modbus.

3.3 Esquema de conexiones con asignación de pines

3.3.1 Asignación de pines

Directrices de cableado

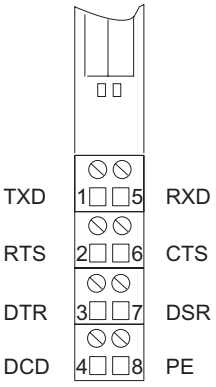
Los cables (bornes 1 a 8) tienen que estar apantallados y la pantalla debe colocarse en ambos lados. Utilice para ello elementos de contacto de pantalla. Encontrará información sobre estos elementos en el apartado *Accesorios* del manual *Sistema de periferia descentralizada ET 200S*.

Asignación de pines para la comunicación RS232C

Con un sistema esclavo puede configurarse un enlace punto a punto. No se admiten canales auxiliares de la interfaz RS232C.

La tabla siguiente muestra la asignación de pines del módulo interfaz serie ET 200S Modbus/USS con un protocolo de comunicación RS232C ajustado.

Tabla 3- 3 Asignación de pines para la comunicación RS232C

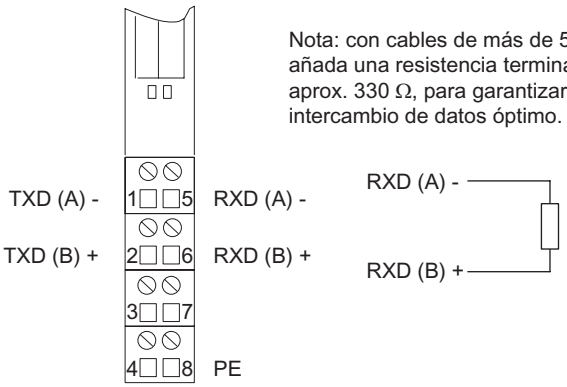
Vista	Observaciones		
	Modo: Dúplex		
	Bornes		
	1	TXD	Datos enviados
	5	RXD	Datos recibidos
	2	RTS	Petición de envío
	6	CTS	Listo para enviar
	3	DTR	Terminal de datos listo
	7	DSR	Registro listo
	4	DCD	Detección del soporte de datos
	8	PE	Tierra

Asignación de pines para la comunicación RS422

Con un sistema esclavo puede configurarse un enlace punto a punto.

La tabla siguiente muestra la asignación de pines del módulo interfaz serie ET 200S Modbus/USS con un protocolo de comunicación RS422 ajustado.

Tabla 3- 4 Asignación de pines para la comunicación RS422

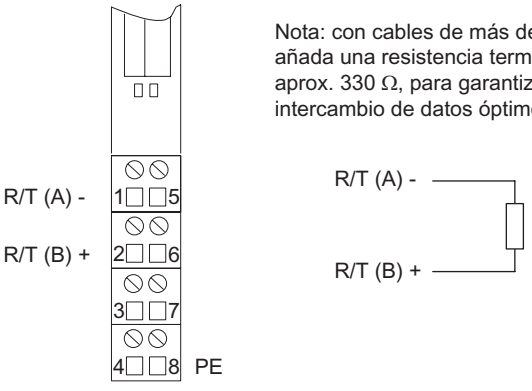
Vista	Asignación de pines	Observaciones	
		Modo: Dúplex	
		Bornes	
		1	TXD (A)-
		5	RXD (A)-
		2	TXD (B)+
		6	RXD (B)+
		8	PE tierra

Asignación de pines para la comunicación RS485

Con un sistema maestro se puede configurar un enlace multipunto (red) con un máximo de 32 esclavos. El driver del módulo conmuta la línea de recepción de 2 hilos entre el envío y la recepción.

La tabla siguiente muestra la asignación de pines del módulo interfaz serie ET 200S Modbus/USS con un protocolo de comunicación RS485 ajustado.

Tabla 3- 5 Asignación de pines para la comunicación RS485

Vista	Asignación de pines	Observaciones	
		Modo: Dúplex	
		Bornes	
		1	R/T (A)-
		2	R/T (B)+
		8	PE tierra

Asignación de pines del cable de conexión RS232C para conector macho de 9 polos

La siguiente figura muestra las conexiones de cables para la comunicación punto a punto según RS232C entre el módulo y un esclavo interlocutor con un conector hembra D de 9 polos.

- En el lado del ET 200S se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el esclavo de comunicación un conector hembra SubD de 9 polos.

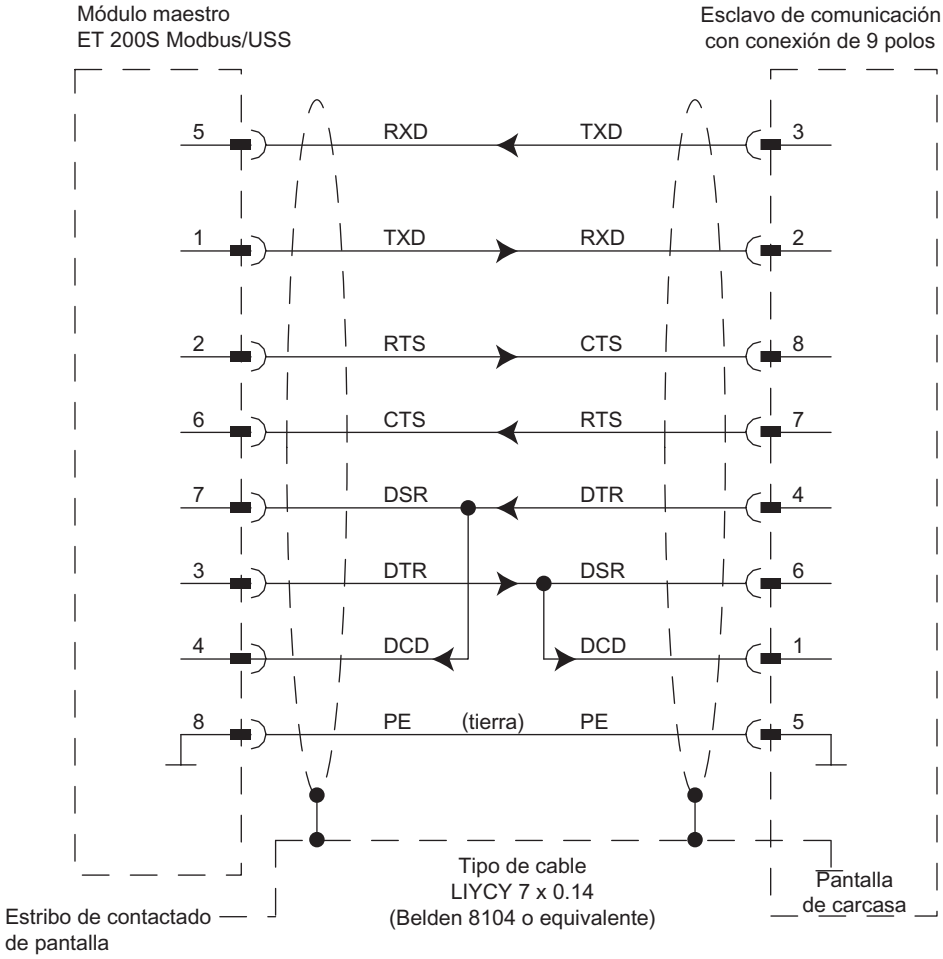


Figura 3-2 Cable de conexión RS232C para conector macho de 9 polos (sistema 1 maestro, 1 esclavo)

Asignación de pines del cable de conexión RS232C para conector macho de 25 polos

La siguiente figura muestra las conexiones de cables para la comunicación punto a punto según RS232C entre el módulo y un esclavo interlocutor con un conector macho D de 25 polos.

- En el lado del ET 200S se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el esclavo de comunicación un conector macho SubD de 25 polos.

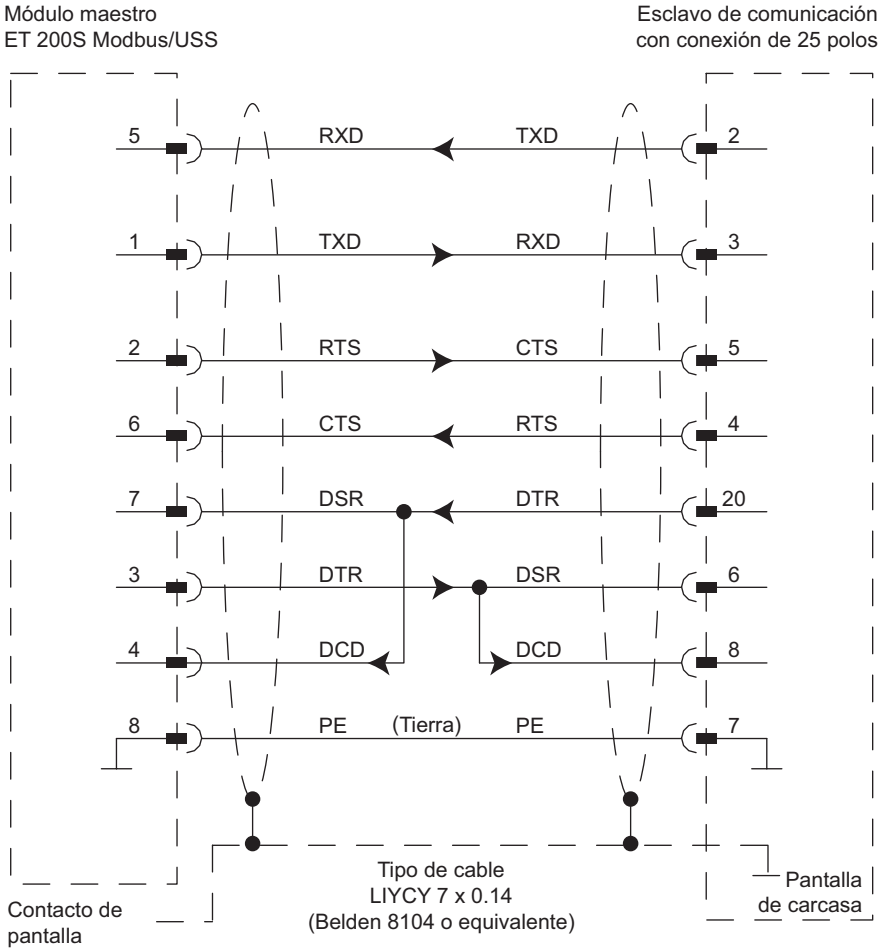


Figura 3-3 Cable de conexión RS232C para conector macho de 25 polos (sistema 1 maestro, 1 esclavo)

Asignación de terminales del cable de conexión RS422 para conectores macho de 15 polos

La siguiente figura muestra las conexiones de cables para la comunicación según RS422 entre el módulo y un esclavo interlocutor con un conector macho D de 15 polos.

- En el lado del ET 200S se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el esclavo de comunicación un conector macho SubD de 15 polos.

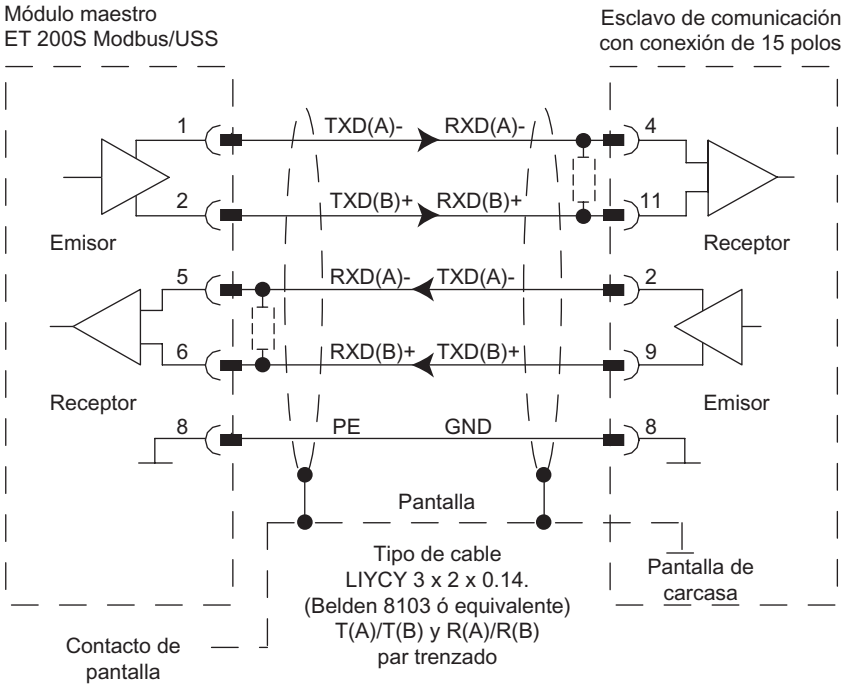


Figura 3-4 Cable de conexión RS422 para conector macho de 15 polos (sistema 1 maestro, 1 esclavo)

Nota

Con cables de más de 50 m añada una resistencia terminadora de aprox. 330 Ω, tal y como se representa en la imagen anterior, para garantizar un intercambio de datos óptimo.

Este tipo de cable puede tener una longitud máxima de 1200 m a 38.400 baudios.

- máx. 1200 m a 19.200 baudios
- máx. 500 m a 38.400 baudios
- máx. 250 m a 76.800 baudios

Asignación de terminales del cable de conexión RS485 para conectores macho de 15 polos

La siguiente figura muestra las conexiones de cables para la comunicación según RS485 entre el módulo y un esclavo interlocutor con un conector macho D de 15 polos.

- En el lado del ET 200S se conectan los hilos de señales a los bornes numerados en correspondencia.
- Utilice en el esclavo de comunicación un conector macho SubD de 15 polos.

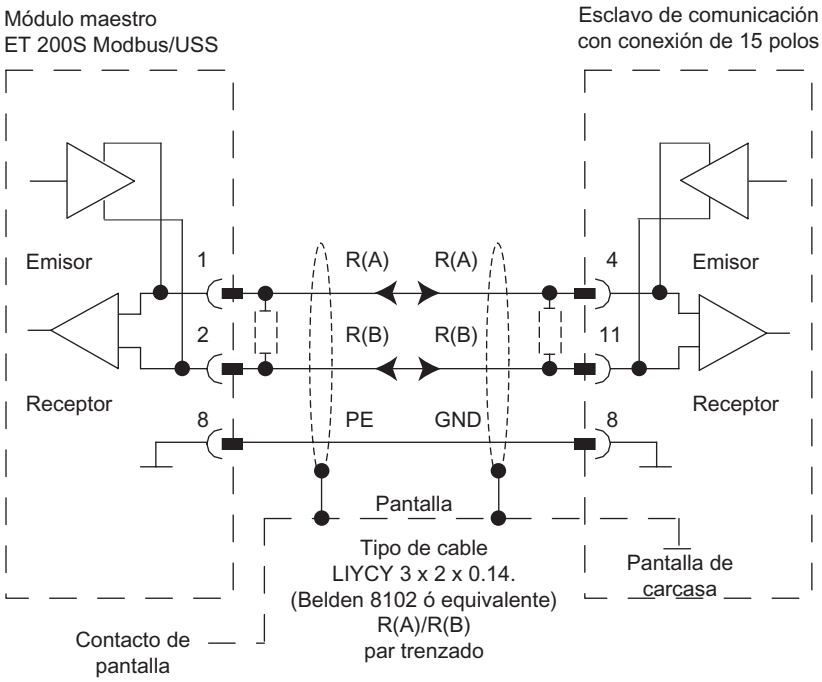


Figura 3-5 Cable de conexión RS485 para conector macho de 15 polos (sistema 1 maestro, 1 esclavo)

Nota

Con cables de más de 50 m añada una resistencia terminadora de aprox. 330 Ω, tal y como se representa en la imagen anterior, para garantizar un intercambio de datos óptimo.

Este tipo de cable puede tener una longitud máxima de 1200 m a 38.400 baudios.

- máx. 1200 m a 19.200 baudios
- máx. 500 m a 38.400 baudios
- máx. 250 m a 76.800 baudios
- máx. 200 m a 115.200 baudios

3.3.2 Interfaz RS-232C

Propiedades de la interfaz RS-232C

La interfaz RS 232C es una interfaz de tensión cuya función es la transmisión serie de datos de acuerdo con la norma RS 232C. La tabla siguiente muestra las propiedades de RS232C.

Tabla 3- 6 Señales de la interfaz RS-232C

Propiedad	Descripción
Tipo	Interfaz de tensión
Conector frontal	Enchufe de bornes estándar de 8 polos del ET 200S
Señales RS 232C	TXD, RXD, RTS, CTS, DTR, DSR, DCD, GND
Velocidad de transmisión	Máximo 115,2 Kbaudios
Longitud de cable	Máximo 15 m, tipo de cable LIYCY 7 x 0.14
Normas	DIN 66020, DIN 66259, EIA-RS 232C, CCITT V.24/V.28
Clase de protección	IP 20

Señales RS 232C

El módulo Modbus/USS soporta las señales RS232C.

Tabla 3- 7 Señales de la interfaz RS-232C

Señal	Designación	Significado
TXD	Datos enviados	Datos de envío; en estado de reposo la línea de envío se mantiene en "1" lógico.
RXD	Datos recibidos	Datos de recepción; el interlocutor debe mantener la línea de recepción en "1" lógico.
RTS	Petición de envío	ON: El módulo está listo para enviar. OFF: El módulo no envía.
CTS	Listo para enviar	El interlocutor puede recibir datos del ET 200S. El módulo interfaz serie espera esta señal como respuesta a RTS = ON.
DTR	Terminal de datos listo	ON: El módulo está conectado y listo para funcionar. OFF: El módulo no está ni conectado ni listo para funcionar.
DSR	Registro listo	ON: El interlocutor está conectado y listo. OFF: El interlocutor no está ni conectado ni listo.
DCD	Detección del soporte de datos	Señal portadora en caso de que se conecte un módem.

Manejo automático de las señales cualificadoras

El manejo automático de las señales cualificadoras RS232C en el módulo está implementado del modo siguiente:

- En cuanto el módulo se ha pasado a un modo de manejo automático de las señales cualificadoras RS232C mediante la parametrización, ajusta las líneas RTS a OFF y DTR a ON (módulo listo para funcionar).

  Sólo es posible enviar y recibir telegramas tras poner la línea DTR a ON. Mientras DTR permanece en OFF, no se pueden recibir datos vía la interfaz RS 232C. Cualquier petición de envío se interrumpe con el correspondiente mensaje de error.
- Si está pendiente una petición de envío, RTS pasa al estado ON y se inicia el tiempo de espera de salida de datos parametrizado. Una vez transcurrido el tiempo de salida de datos, y con CTS = ON, los datos se envían a través de la interfaz RS 232C.
- Si durante el envío la línea CTS no pasa a ON durante el tiempo de espera, o si durante el proceso de salida, CTS cambia al estado OFF, se interrumpe el envío mostrando el mensaje correspondiente.
- Tras la emisión de los datos y una vez transcurrido el tiempo de RTS a OFF parametrizado, la línea RTS pasa al estado de inactividad. El ET 200S no espera a que CTS pase a OFF.
- Es posible recibir datos vía la interfaz RS 232C, tan pronto como se active la línea DSR (ON) . Si el búfer de recepción del módulo está a punto de desbordarse, el módulo no reacciona.
- Al cambiar de DSR = ON a OFF se interrumpe tanto el envío en curso como la recepción de datos con un mensaje de error.

Nota

El manejo automático de las señales cualificadoras de RS-232C sólo es posible en modo semidúplex.

Nota

"Tiempo hasta RTS OFF" debe ajustarse en la interfaz de parametrización de modo que el interlocutor pueda recibir por completo los últimos caracteres del telegrama antes de que se retire RTS y, con ello, la petición de envío. El "Tiempo de espera salida de datos" debe estar ajustado para que el interlocutor pueda estar listo para recibir antes de que se agote el tiempo.

Cronograma de las señales cualificadoras

La siguiente figura muestra el desarrollo temporal de una petición de envío:

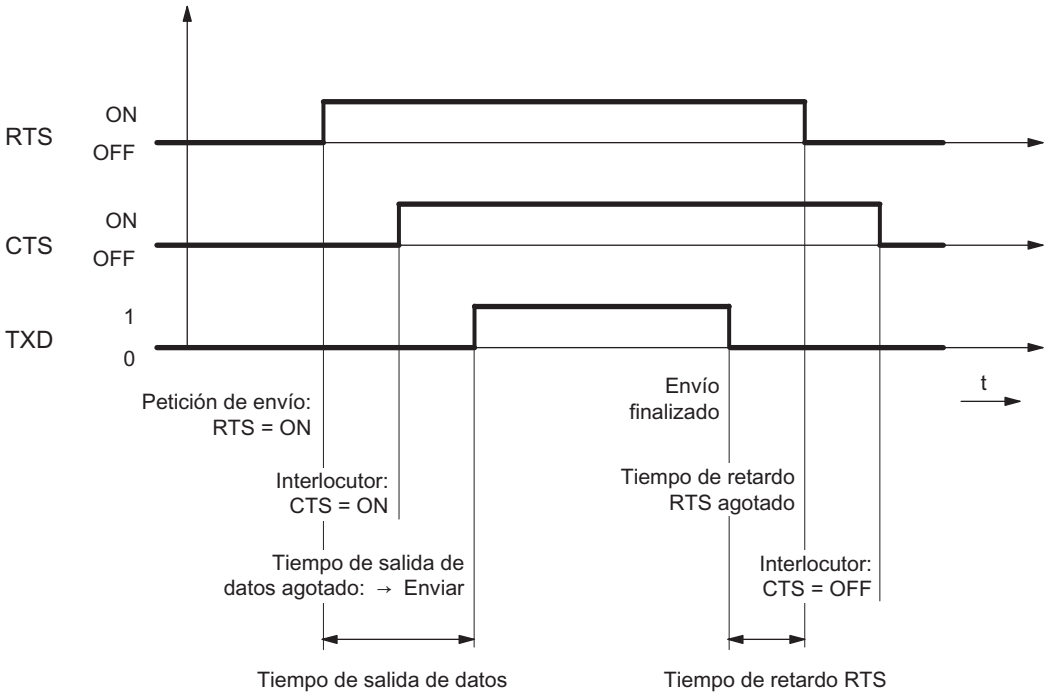


Figura 3-6 Cronograma para el manejo automático de las señales cualificadoras RS 232C

3.3.3 Interfaz RS-422/485

Propiedades de la interfaz RS-422/458

La interfaz RS-422/485 es una interfaz diferencial y sirve para transferir datos en serie según la norma RS422/485. La tabla muestra las propiedades de la interfaz RS-422/485.

Tabla 3- 8 Propiedades de la interfaz RS-422/485

Propiedad	Descripción
Tipo	Interfaz diferencial
Conector frontal	Enchufe de bornes estándar de 8 polos del ET 200S
Señales RS-422	TXD (A)-, RXD (A)-, TXD (B)+, RXD (B)+, GND
Señales RS-485	R/T (A)-, R/T (B)+, GND
Velocidad de transmisión	Máximo 115,2 Kbaudios
Longitud de cable	Máximo 1200 m, tipo de cable LIYCY 7 x 0.14
Normas	EIA RS-422/485, CCITT V.11/V.27
Clase de protección	IP 20

3.4 Protocolo de transmisión Modbus

3.4.1 Propiedades y estructura del telegrama

Propiedades

El procedimiento utilizado para la transferencia Modbus es un procedimiento semidúplex asíncrono y con transparencia de códigos. La transferencia de datos se realiza sin handshake.

El módulo lanza la transferencia (en calidad de maestro). Tras la salida del telegrama de petición, el módulo espera un telegrama de respuesta del esclavo durante el tiempo de supervisión de respuesta.

Estructura del telegrama

El intercambio de datos "maestro-esclavo" o "esclavo-maestro" empieza con la dirección del esclavo, seguida por el código de función. A continuación, se transfieren los datos. El intercambio de datos "maestro-esclavo" o "esclavo-maestro" dispone de los elementos siguientes:

DIRECCIÓN DE ESCLAVO	Dirección de esclavo de Modbus
CÓDIGO DE FUNCIÓN	Código de función de Modbus
Datos	Datos de telegrama: Byte_Count, Coil_Number, Data
COMPROBACIÓN CRC	Suma de comprobación del telegrama

La configuración del campo de datos depende del código de función empleado. Al final del telegrama se transfiere la comprobación CRC. La tabla muestra los componentes de la estructura del telegrama.

Tabla 3- 9 Estructura del telegrama

Dirección	Función	Datos	COMPROBACIÓN CRC
Byte	Byte	n bytes	2 bytes

3.4.2 Dirección de esclavo

Descripción

La dirección del esclavo puede estar entre 1 y 247. La dirección da acceso a un esclavo concreto del bus.

Telegrama de transferencia

Con la dirección de esclavo cero, el maestro accede a todos los esclavos del bus.

Nota

Los telegramas de transferencia sólo están permitidos en combinación con los códigos de función 05, 06, 15 ó 16.

En un telegrama de transferencia, el esclavo no envía ningún telegrama de respuesta.

3.4.3 Códigos de función de maestro y esclavo

Códigos de función de maestro y esclavo

El código de función define el significado y la estructura del telegrama. La tabla siguiente lista los códigos de función así como su disponibilidad para maestro y esclavos.

Tabla 3- 10 Códigos de función de maestro y esclavo

Código de función	Descripción	Maestro	Esclavo
01	Read Coil Status	√	√
02	Read Input Status	√	√
03	Read Holding Registers	√	√
04	Read Input Registers	√	√
05	Force Single Coil	√	√
06	Preset Single Register	√	√
07	Read Exception Status	√	-
08	Loop Back Test	√	√
11	Fetch Communications Event Counter	√	-
12	Fetch Communications Event Log	√	-
15	Force Multiple Coils	√	√
16	Preset Multiple Registers	√	√

3.4.4 Campo de datos DATA

Descripción

En el campo de datos DATA se transfieren los datos específicos del código de función:

- Byte count
- Coil Start Address
- Register Start Address
- Number of Coils
- Number of Registers

3.4.5 Fin de telegrama y comprobación CRC

Descripción

El telegrama termina con la suma de comprobación CRC-16 de 2 bytes. Se calcula siguiendo el polinomio siguiente:

$x^{16} + x^{15} + x^2 + 1$

Primero se transfiere el byte Low y después el byte High.

Detección de fin de telegrama.

El módulo Modbus/USS reconoce el fin del telegrama cuando no se realiza ninguna transferencia en el tiempo que tardan en transferirse tres caracteres y medio (3,5 veces el tiempo de retardo entre caracteres).

Este timeout de fin de telegrama depende de la velocidad de transferencia.

Cuando se agota el timeout de fin de telegrama, se evalúa el telegrama de respuesta recibido del esclavo y se comprueba formalmente.

Tabla 3- 11 Fin del telegrama

Velocidad de transmisión	Timeout
115.200 bps	1 ms
76.800 bps	1 ms
57.600 bps	1 ms
38.400 bps	1 ms
19.200 bps	2 ms
9.600 bps	4 ms
4.800 bps	8 ms
2.400 bps	16 ms
1.200 bps	32 ms
600 bps	65 ms
300 bps	130 ms
115 bps	364 ms

3.4.6 Respuestas de excepción

Telegrama de respuesta en caso de error

Si el esclavo detecta un error en el telegrama de petición del maestro (p. ej. dirección de registro no válida), el esclavo ejecuta las acciones siguientes:

- El esclavo activa el bit más significativo en el código de función del telegrama de respuesta.
- El esclavo envía un byte de código de error (código de excepción) para describir la causa del error.

Ejemplo: Telegrama de código de excepción

El telegrama de respuesta de código de error del esclavo puede tener, por ejemplo, la siguiente estructura: Dirección de esclavo 5, código de función 5, código de excepción 2.

Telegrama de respuesta del esclavo EXCEPTION_CODE_xx	05H	Dirección del esclavo
	85H	Código de función
	02H	Código de excepción (1 a 7)
	xxH	Código de comprobación CRC "Low"
	xxH	Código de comprobación CRC "High"

Cuando se recibe un telegrama de respuesta con código de error del driver, se finaliza la petición actual con errores.

Asimismo, en el área SYSTAT se registra un número de error que corresponde al código de error recibido (código de excepción 1-7).

No se realiza ninguna entrada en un bloque de datos de destino S_RCV.

Tabla de códigos de errores

La tabla lista los códigos de error que envía el módulo.

Tabla 3- 12 Códigos de error

Código de excepción	Descripción	Causa posible
01	Función no permitida	Se ha recibido un código de función no permitido.
02	Dirección de datos no permitida	Acceso a un área SIMATIC que no está habilitada (véase Tabla de conversión de datos Modbus)
03	Valor de datos no permitido	Longitud superior a 2040 bits o 127 registros, campo de datos no FF00 ó 0000 para FC05, subcódigo de diagnóstico <> 0000 para FC08.
04	Fallo en el equipo correspondiente	Inicialización mediante FB de comunicación Modbus todavía no ejecutada o FB notifica errores. Error en la transferencia de datos módulo – CPU (ejemplo: DB no disponible, longitud máxima transmisible superada (tamaño de bloque CPU <-> módulo).

3.5 Driver maestro Modbus

3.5.1 Empleo del driver maestro Modbus

Finalidad de uso

El driver Modbus ET 200S puede utilizarse en los sistemas de automatización S7 y puede establecer enlaces de comunicación en serie con sistemas interlocutores.

Con este driver es posible un enlace de comunicación entre el driver maestro Modbus ET 200S y sistemas de automatización aptos para Modbus.

Proceso de transferencia

Para la transferencia, el protocolo Modbus se utiliza en formato RTU. La transferencia de datos se desarrolla siguiendo el principio maestro-esclavo.

El maestro lanza la transferencia.

El maestro Modbus puede emplear los códigos de función 01, 02, 03, 04, 05, 06, 07, 08, 11, 12, 15 y 16.

Interfaces y protocolos aplicables

Para el módulo pueden emplearse las interfaces RS-232 o RS-422/485 (X27).

La interfaz RS422/485 puede utilizarse con este driver tanto en modo de 2 hilos como en modo de 4 hilos. En modo de 2 hilos es posible conectar un máximo de 32 esclavos a un maestro en modo semidúplex. De este modo se crea un enlace multipunto (red). En modo de 4 hilos (RS-422) sólo se admiten 1 maestro y 1 esclavo en modo semidúplex.

3.5.2 Transferencia de datos con el maestro de Modbus ET 200S

Introducción

La transmisión de datos entre el módulo y la CPU se realiza mediante los FBs S_SEND y S_RCV. El FB S_SEND se activa con un flanco en la entrada REQ cuando deben emitirse datos. El FB S_RCV se prepara para la recepción con EN_R=1. En todos los códigos de función de lectura es necesario un S_RCV.

FB3 S_SEND: Enviar datos a un interlocutor

Para ejecutar una petición del maestro Modbus tienen que activarse los FBs S_SEND y S_RCV. El FB S_SEND se activa con un flanco en la entrada REQ cuando deben emitirse datos al módulo. El FB S_RCV se prepara con EN_R=1 para recibir datos del módulo. En todos los códigos de función de lectura es necesario un S_RCV. La figura siguiente muestra el comportamiento global de los parámetros S_SEND y S_RCV cuando se ejecuta una petición Modbus.

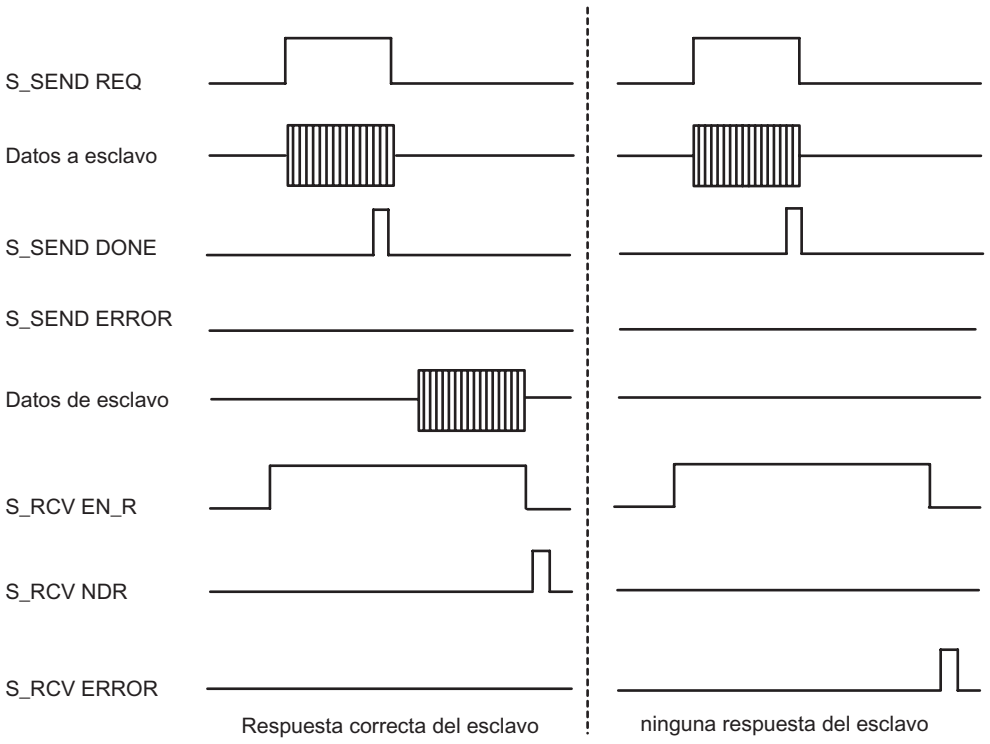


Figura 3-7 Cronograma de una petición Modbus

Con flanco positivo en la entrada REQ se inicia la transmisión de los datos. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

El bloque de función FB S_SEND se puede llamar en el ciclo con el estado de señal "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al módulo y el FB S_SEND pasa al estado básico. Los datos que ya ha recibido el módulo se envían al interlocutor. Si hay un estado de señal "1" estático en la entrada R, la transmisión está desactivada.

En el parámetro LADDR se indica la dirección del módulo interfaz serie ET 200S Modbus/USS con el que debe realizarse la conexión.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número correspondiente de evento se indica en STATUS. Si no se produce ningún fallo, STATUS tiene el valor 0. DONE y ERROR/STATUS se indican también en caso de RESET del FB S_SEND. Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Petición de lectura del maestro Modbus

Dado que la interfaz entre el programa de usuario y el módulo interfaz funciona en modo semidúplex, hay que tener en cuenta lo siguiente:

Después de una petición de lectura del maestro Modbus con acuse positivo, hay que recoger los datos recibidos del módulo interfaz con el bloque de función S_RCV antes de iniciar una nueva petición de envío del maestro Modbus.

Arranque

El parámetro COM_RST del FB S_SEND permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S Modbus/USS (número de bytes en el área de periferia, ya sea en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

Una vez que el FB ha proporcionado información sobre el módulo ET 200S Modbus/USS, el parámetro COM_RST se desactiva automáticamente.

La tabla siguiente muestra las representaciones de AWL y KOP del FB3 S_SEND.

Nota

La entrada REQ está activada por flanco. Es suficiente un flanco positivo en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure la petición de recepción, el parámetro EN_R debe configurarse con el RLO "1" (resultado lógico).

Nota

El bloque de función S_SEND no tiene comprobación de parámetros. Si hay parámetros no válidos, la CPU pasa al estado operativo STOP.

Antes de que el módulo pueda procesar una petición lanzada después de que la CPU pase de STOP a RUN, debe haber concluido el mecanismo de arranque de la CPU de ET 200S del FB S_SEND. Una petición iniciada entre tanto no se pierde. Se transfiere al módulo una vez concluida la coordinación de arranque.

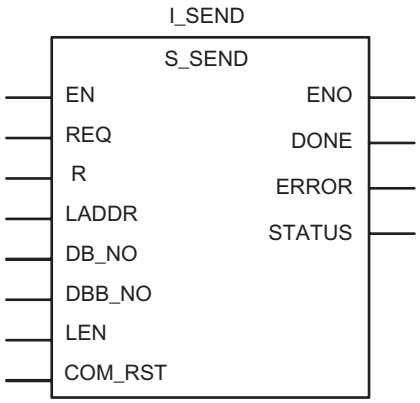
Llamada del FB3

La tabla muestra las representaciones AWL y KOP del FB3 S_SEND.

Representación AWL

```
CALL               S_SEND, I_SEND
     REQ:           =
     R:             =
     LADDR:         =
     DB_NO:         =
     DBB_NO:        =
     LEN:           =
     DONE:          =
     ERROR:         =
     STATUS:        =
     COM_RST:       =
```

Representación KOP



Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha completado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Asignación en el área de datos

El FB S_SEND opera conjuntamente con un DB de instancia I_SEND. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: En caso de error, STATUS == W#16#1E0F, puede consultar más detalles sobre el error en la variable SFCERR. Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Parámetros del FB3 S_SEND

La tabla lista los parámetros de S_SEND (FB3).

Tabla 3- 13 FB3: Parámetros de S_SEND

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
REQ	INPUT	BOOL	Inicio de petición con flanco positivo	
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección básica de la interfaz serie ET 200S	La dirección básica se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos	Nº de DB de envío: Específico de la CPU (no se permite el valor 0)
DBB_NO	INPUT	INT	Número del byte de datos	0 ≤ DBB_NO ≤ 8190 Datos transferidos por palabra de datos
LEN	INPUT	INT	Longitud de datos	1 ≤ LEN ≤ 224 indicación en número de bytes
DONE [1]	OUTPUT	BOOL	La petición ha finalizado sin errores	Parámetro STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, el parámetro STATUS contiene la información de error.
COM_RST	IN_OUT	BOOL	Rearranque del FB	

[1] Tras una petición de envío correcta, estos parámetros están disponibles durante **un** ciclo de la CPU.

Cronograma de FB3 S_SEND

La figura siguiente muestra el comportamiento de los parámetros DONE y ERROR según sea el cableado de las entradas REQ y R.

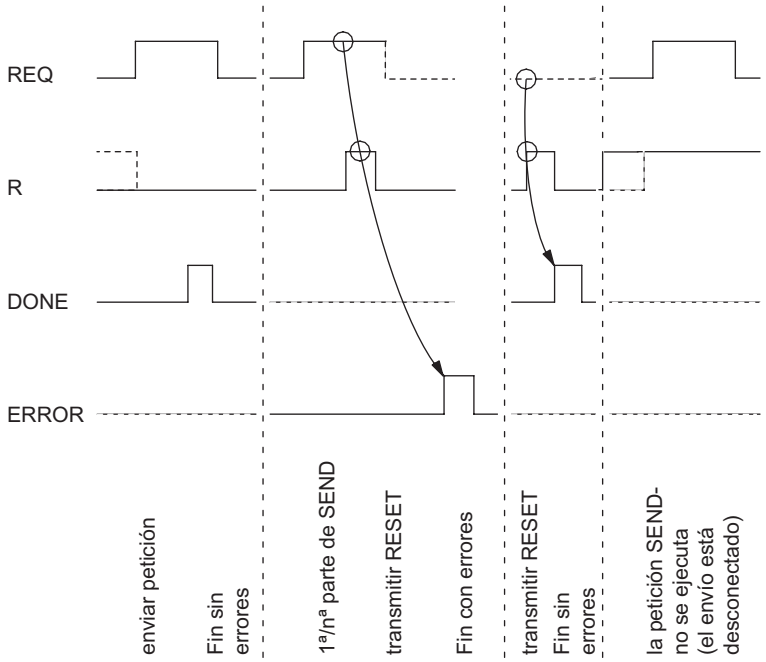


Figura 3-8 Cronograma de FB3 S_SEND

Nota

La entrada REQ está activada por flanco. Es suficiente un flanco positivo en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

FB2 S_RCV: Recibir datos de un interlocutor

El FB S_RCV transmite datos del módulo a un área de datos S7 especificado por los parámetros DB_NO, DBB_NO. Para la transmisión de datos, el FB S_RCV se llama cíclicamente o bien estáticamente (sin condiciones) en un programa controlado por tiempo.

Con el estado (estático) de señal "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos de la interfaz serie. Una transmisión en curso se puede cancelar con el estado de señal "0" en el parámetro EN_R. La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desconectada mientras exista el estado de señal "0" en el parámetro EN_R. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Si el bloque de función detecta el estado de señal "1" en el parámetro R, se cancela la petición actual de transmisión y el FB S_RCV pasa al estado inicial. La recepción estará desactivada mientras el parámetro R tenga el estado lógico "1". Cuando el estado de señal vuelve a ser "0", el telegrama interrumpido se vuelve a recibir desde el principio.

En el parámetro LADDR se indica el módulo interfaz serie ET 200S Modbus/USS con el que debe realizarse la conexión.

La salida NDR indica "Petición terminada sin error/Datos aceptados" (leídos todos los datos). ERROR indica que se ha producido un error. Si se ha producido un error, el número de error correspondiente se muestra en STATUS cuando el búfer de recepción está lleno en más de 2/3 partes. Tras cada llamada de S_RCV, STATUS contiene una advertencia si ERROR no está activado. Si no se han producido errores ni advertencias, STATUS tiene el valor 0.

NDR y ERROR/STATUS también se emiten en caso de RESET del FB S_RCV (parámetro LEN == 16#00). Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Arranque

El parámetro COM_RST del FB S_RCV permite notificar un arranque al FB.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame el FB en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- el FB proporciona información sobre el módulo ET 200S Modbus/USS (número de bytes en el área de periferia, ya sea en periferia descentralizada o no).
- el FB se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes del último cambio a Stop de la CPU).

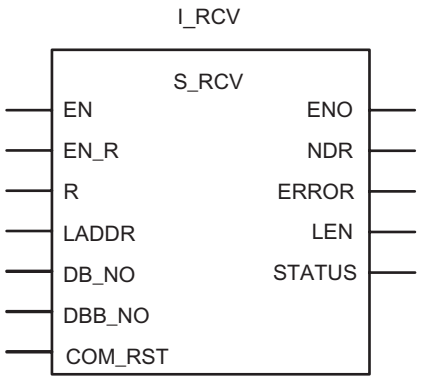
Una vez que el FB ha proporcionado información sobre el módulo ET 200S Modbus/USS, el parámetro COM_RST se desactiva automáticamente.

Nota

El bloque de función S_RCV no tiene comprobación de parámetros. Si hay parámetros no válidos, la CPU puede pasar al estado operativo STOP.

Antes de que el módulo pueda recibir una petición después de que la CPU pase de STOP a RUN, debe haber concluido el mecanismo de arranque de la CPU del ET 200S del FB S_RCV.

La tabla muestra las representaciones de AWL y KOP del FB2 S_RCV.

Representación AWL		**Representación KOP**
CALL	S_RCV, I_RCV	
EN_R:	=	
R:	=	
LADDR:	=	
DB_NO:	=	
DBB_NO:	=	
NDR:	=	
ERROR:	=	
LEN:	=	
STATUS:	=	
COM_RST:	=	

Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha completado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Asignación en el área de datos

El FB S_RCV opera conjuntamente con un DB de instancia I_RCV. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

La tabla lista los parámetros del FB2 S_RCV.

Nota

Excepción: En caso de error, STATUS == W#16#1E0D, puede consultar más detalles sobre el error en la variable SFCERR . Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Tabla 3- 14 FB2: Parámetros de S_RCV

Nombre	Tipo	Tipo de datos	Descripción	Valores permitidos, observación
EN_R	INPUT	BOOL	Habilitar lectura de datos	
R	INPUT	BOOL	Interrupción de la petición	Se interrumpe la petición en curso. Bloqueo de recepción.
LADDR	INPUT	INT	Dirección básica de la interfaz serie ET 200S	La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos	N.º DB de recepción: Específico de la CPU (no se permite el valor 0)
DBB_NO	INPUT	INT	Número del byte de datos	0 ≤ DBB_NO ≤ 8190 Datos recibidos por palabra de datos
NDR[1]	OUTPUT	BOOL	La petición ha finalizado sin errores, datos transferidos	Parámetro STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	La petición ha finalizado con errores	El parámetro STATUS contiene la información de error.
LEN[1]	OUTPUT	INT	Longitud del telegrama recibido	1 ≤ LEN ≤ 224 indicación en número de bytes
STATUS [1]	OUTPUT	WORD	Especificación del error	Si ERROR == 1, el parámetro STATUS contiene la información de error.
COM_RST	IN_OUT	BOOL	Rearranque del FB	

[1] Tras una petición de recepción correcta, estos parámetros están disponibles durante un ciclo de la CPU.

Cronograma de FB2 S_RCV

La figura siguiente muestra el comportamiento de los parámetros NDR, LEN y ERROR según sea el cableado de las entradas EN_R y R.

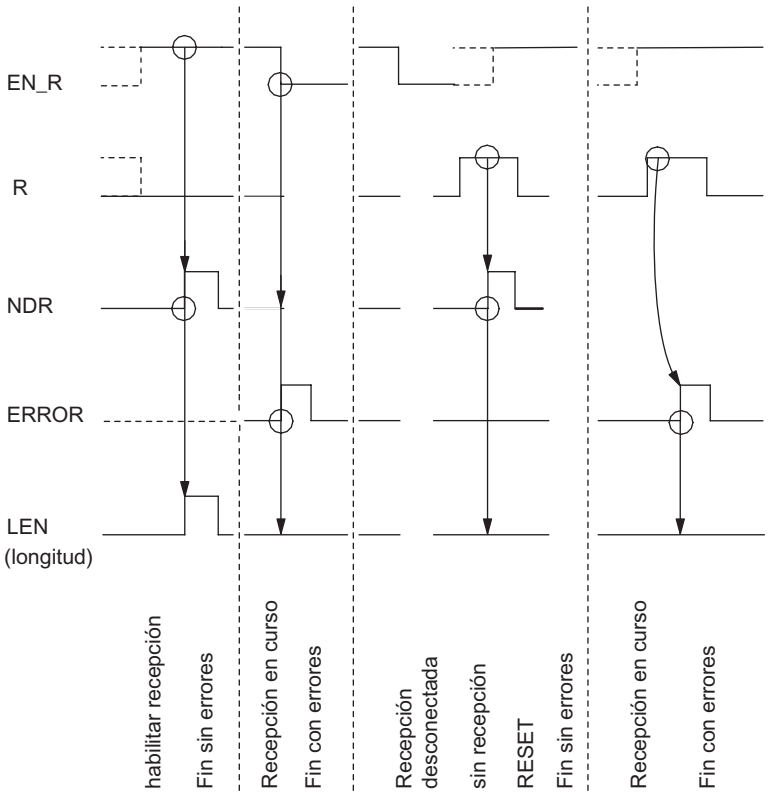


Figura 3-9 Cronograma del FB2 S_RCV

Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure la petición de recepción, el parámetro EN_R debe configurarse con el RLO "1" (resultado lógico).

3.5.3 Configuración y parametrización del maestro Modbus

Configuración del módulo Modbus

Si se comunica con un maestro S7 utilizando el módulo a través de una red PROFIBUS, trabaje en la configuración de hardware en STEP 7 para ajustar el módulo en la red PROFIBUS y ajustar los parámetros de comunicación del módulo.

Si selecciona el maestro Modbus en el catálogo de hardware y lo inserta en el ET 200S básico en la configuración de la red, la referencia del módulo, el número del slot y las direcciones de entradas y salidas se incluirán automáticamente en la tabla de configuración. A continuación, se puede abrir el cuadro de diálogo de propiedades del maestro Modbus y ajustar el tipo de comunicación y otros parámetros.

Parametrizar el driver maestro

La tabla lista los parámetros que pueden ajustarse para el driver Modbus del módulo.

Tabla 3- 15 Parámetros del driver maestro Modbus

Parámetro	Descripción	Rango de valores	Valor por defecto
Alarma de diagnóstico	Indique si el módulo crea una alarma de diagnóstico cuando surge un error grave.	• No • Sí	No
Activar reconocimiento BREAK	Si se produce una ruptura de la línea o no se ha conectado ningún cable de interfaz, el módulo notifica el mensaje de error "Break".	• No • Sí	No
Tipo de interfaz	Indique la interfaz eléctrica que debe emplearse.	• RS-232C • RS-422 (dúplex) • RS-485 (semidúplex)	RS-232C
Preajuste dúplex y semidúplex de la línea de recepción	Indique el preajuste de la línea de recepción en los modos de operación RS-422 y RS-485. No en el modo de operación RS-232C. La configuración "Nivel invertido" sólo es requerida en caso de repuesto para asegurar la compatibilidad.	RS422: R(A) 5V / R(B) 0V (BREAK) R(A) 0V / R(B) 5V Nivel invertido RS485: Ninguno R(A) 0V / R(B) 5V	RS422: R(A) 5V / R(B) 0V (BREAK) RS485: R(A) 0V / R (B) 5V

Parámetro	Descripción	Rango de valores	Valor por defecto
Control de flujo de datos (con parámetros predeterminados; cambiar valores predeterminados en el programa de usuario)	Se pueden enviar y recibir datos con control del flujo de datos. Mediante el control del flujo de datos, la transmisión de datos se sincroniza cuando un interlocutor trabaja más rápidamente que el otro. Seleccione el tipo de control del flujo de datos y ajuste los parámetros correspondientes. Nota: Con la interfaz RS 485 no es posible el control de flujo de datos. El control del flujo de datos con "Manejo automático de las señales V24" sólo es posible con la interfaz RS232C.	• Sin • Manejo automático de las señales V.24	Sin
Velocidad de transmisión	Seleccione la velocidad de la transmisión de datos en bits por segundo.	• 110 • 300 • 600 • 1.200 • 2.400 • 4.800 • 9.600 • 19.200 • 38.400 • 57.600 • 76.800 • 115.200	9600
Bits de parada	Seleccione la cantidad de bits de parada que se colocan detrás de cada carácter durante la transferencia y que marcan el fin de un carácter.	• 1 • 2	1
Paridad	La secuencia de los bits de datos puede ampliarse en un carácter para incluir el bit de paridad. El valor adicional (0 ó 1) traslada el valor de todos los bits (bits de datos y bit de paridad) a un estado definido. **Sin**: Los datos se envían sin bit de paridad. **Impar**: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es impar con el estado de señal "1". **Par**: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es par con el estado de señal "1".	• Sin • Impar • Par	Par
Tiempo de reacción	Tiempo permitido para la respuesta del esclavo.	de 50 ms a 655.000 ms	2000 ms

Parámetro	Descripción	Rango de valores	Valor por defecto
Modo de operación	"Modo normal" "Supresión de frecuencias perturbadoras"	• Normal • Supresión de frecuencias perturbadoras	Normal
Multiplicador de retardo de caracteres	Utiliza un multiplicador del tiempo de retardo entre caracteres de 1-10.	1 a 10	1
Borrar búfer de recepción de la interfaz serie al arrancar	Indique si el búfer de recepción del módulo interfaz serie debe borrarse automáticamente cuando la CPU pasa del estado operativo STOP a RUN (arranque de la CPU). Ello permite asegurarse de que el búfer de recepción de la interfaz serie sólo recibe telegramas recibidos después de arrancar la CPU.	• No • Sí	Sí

- **Dúplex (RS422), modo a cuatro hilos**

  En este modo de operación se envían datos a través de la línea de envío T(A), T(B) y se reciben a través de la línea de recepción R(A), R(B). El tratamiento de errores se produce según la función configurada con el parámetro “Modo de operación del driver" (normal o supresión de frecuencias perturbadoras).

- **Semidúplex (RS485), modo a dos hilos**

  En este modo de operación, el driver ejecuta un modo de conmutación de la línea de recepción de 2 hilos R(A), R(B) de la interfaz entre modo de envío y de recepción. El inicio de un telegrama de recepción es reconocido a través de la dirección correcta de esclavo. Al utilizar el acoplamiento punto y punto, se recomienda preajustar la preasignación de la línea de recepción con los parámetros
  R (A) 0V, R(B) 5V.

- **Ocupación estándar de la línea de recepción**

  Este parámetro indica el estado base de la línea de recepción en los modos de operación RS-422 y RS-485. No se utiliza para el modo de operación RS232C.

  – **R(A) 5V, R(B) 0V (BREAK)**
    El módulo preajusta la siguiente asignación para la línea de dos hilos R(A), R(B):
    R(A) --> +5V, R(B) --> 0V ($V_A – V_B$ = +0,3 V).
    Esto significa que en el módulo hay nivel BREAK con rotura de hilo.

  – **R(A) 0V, R(B) 5V (High)**
    El módulo preajusta la siguiente asignación para la línea de dos hilos R(A),R(B):
    R(A) --> 0V, R(B) --> +5 V ($V_A – V_B$ = -0,3 V).
    Esto significa que en el módulo hay nivel HIGH con rotura de línea (o bien en estado de reposo si ningún esclavo envía). El estado de línea BREAK no puede detectarse.

  – **Sin (sólo con RS485)**
    En una conexión multipunto, la preasignación de la línea de recepción está desactivada.

- **Velocidad de transferencia**

  La velocidad de transferencia máxima es la velocidad de transferencia de datos en bits por segundo (bps). La velocidad de transferencia máxima del módulo es de 38400 bps en modo semidúplex.

- **Bits de datos**

  La cantidad de bits de datos describe en cuántos bits puede representarse un carácter que debe transferirse. Tienen que configurarse siempre 8 bits de datos. Hay que emplear siempre una trama de caracteres de 11 bits. Si se ajusta paridad "sin", deben seleccionarse 2 bits de parada.

- **Bits de parada**

  La cantidad de bits de parada define la distancia temporal mínima entre dos caracteres que deben transferirse. Hay que emplear siempre una trama de caracteres de 11 bits. Si se ajusta paridad "sin", deben seleccionarse 2 bits de parada.

- **Paridad**

  El bit de paridad se utiliza para la seguridad de los datos. Complementa la cantidad de bits de datos transferidos en un número par o impar en función de la parametrización. Si se ajusta la paridad "sin" no se transfiere ningún bit de paridad. Esto reduce la seguridad de transmisión. Hay que emplear siempre una trama de caracteres de 11 bits. Si se ajusta paridad "sin", deben seleccionarse 2 bits de parada.

- **Tiempo de respuesta**

  El tiempo de supervisión de respuesta es el tiempo que el maestro espera un telegrama de respuesta del esclavo tras emitir un telegrama de solicitud.

- **Modo normal**

  En este modo de operación, todos los errores de transmisión o BREAK detectados antes y después del telegrama de recepción del esclavo provocan un mensaje de error pertinente.

- **Supresión de frecuencias perturbadoras**

  Si al principio del telegrama de recepción se reconoce BREAK en la línea de recepción o si el bloque de interfaz del módulo determina errores de transmisión, el driver ignora esta recepción defectuosa. El inicio de un telegrama de recepción es reconocido a través de la dirección correcta de esclavo. También se ignoran errores de transmisión o BREAK que aparecen una vez finalizado el telegrama de recepción (código CRC).

- **Multiplicador de retardo de caracteres**

  Cuando un interlocutor no puede cumplir las exigencias en cuanto a tiempo de la especificación Modbus, existe la posibilidad de multiplicar el tiempo de retardo entre caracteres t_{TRC} por el factor de multiplicación f_{MUL}. El tiempo de retardo entre caracteres sólo debe modificarse en aquellos casos en que el interlocutor del acoplamiento no puede mantener el tiempo requerido. El tiempo de retardo entre caracteres modificado t_{TRC} resulta de:

$t_{TRC} = t_{TRC\_TAB} * f_{MUL}$;

t_{TRC_TAB}:	Valor de tabla para t_{TRC}
f_{MUL}:	Factor de multiplicación

Nota

Tenga en cuenta también las indicaciones recogidas en los temas Datos de identificación (Página 63) y Cargar actualizaciones de firmware a posteriori (Página 65).

3.5.4 Códigos de función empleados por el maestro Modbus

Tabla de códigos de función

La tabla lista los códigos de función soportados por el driver maestro Modbus.

Tabla 3- 16 Parámetros del driver maestro Modbus

Código de función	Descripción	Función en SIMATIC S7	
01	Read Output Status	Leer por bits	Marca M
		Leer por bits	Salidas A
		Leer por bits (intervalo de 16 bits)	Temporizadores T
		Leer por bits (intervalo de 16 bits)	Contadores Z
02	Read Input Status	Leer por bits	Marca M
		Leer por bits	Entradas E
03	Read Output Registers	Leer por palabras	Bloque de datos DB
04	Read Input Registers	Leer por palabras	Bloque de datos DB
05	Force Single Coil	Escribir por bits	Marca M
		Escribir por bits	Salidas A
06	Preset Single Register	Escribir por palabras	Bloque de datos DB
07	Read Exception Status	Leer por bits	Estado de 8 bits
08	Loop back diagnostic test	-	-
11	Fetch Communications Event Counter	Leer 2 palabras	Estado de evento y contador
12	Fetch Communications Event Log	Leer 70 bytes	Protocolo de eventos
15	Force Multiple Coils	Escribir por bits (1...2040 bits)	Marca M
		Escribir por bits (1...2040 bits)	Salidas A
16	Preset Multiple Registers	Escribir por palabras (1...127 registros)	Bloque de datos DB

3.5.5 Código de función 01 – Read Output Status

Finalidad y estructura

Función	Esta función permite leer bits individuales del esclavo.
Dirección inicial	El driver no comprueba el parámetro dirección inicial de bit y lo envía sin modificarlo.
Número de bits	Como número de bits (number of coils) se admite cualquier valor entre 1 y 1768.
LEN en bytes	6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#1	Código de función
+2.0	Dirección inicial de bit	WORD	W#16#0040	Dirección inicial de bit
+4.0	Número de bits	INT	16	Número de bits

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#1701	Datos

El driver registra los datos del telegrama de respuesta en el DB de destino por palabras. El primer byte recibido se registra como byte Low de la primera palabra "data[1]", el tercer byte recibido como byte Low de la segunda palabra "data[2]", etc. Si se leen menos de 9 bits o si sólo se ha leído un byte Low, en el byte High restante de la última palabra se registra el valor 00H.

3.5.6 Código de función 02 – Read Input Status

Finalidad y estructura

Función	Esta función permite leer bits individuales del esclavo.
Dirección inicial	El driver no comprueba el parámetro dirección inicial de bit y lo envía sin modificarlo.
Número de bits	Como número de bits (number of coils) se admite cualquier valor entre 1 y 1768.
LEN en bytes	6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#2	Código de función
+2.0	Dirección inicial de bit	WORD	W#16#0120	Dirección inicial de bit
+4.0	Número de bits	INT	24	Número de bits

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#2604	Datos
+2.0	data[2]	WORD	W#16#0048	Datos

El driver registra los datos del telegrama de respuesta en el DB de destino por palabras. El primer byte recibido se registra como byte Low de la primera palabra "data[1]", el tercer byte recibido como byte Low de la segunda palabra "data[2]", etc.

Si se leen menos de 9 bits o si sólo se ha leído un byte Low, en el byte High restante de la última palabra se registra el valor 00H.

3.5.7 Código de función 03 – Read Output Registers

Finalidad y estructura

Función	Esta función permite leer registros individuales del esclavo.
Dirección inicial	El driver no comprueba el parámetro dirección inicial del registro y lo envía sin modificarlo.
Número de bits	Pueden leerse como máximo 110 registros (1 registro = 2 bytes).
LEN en bytes	6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#3	Código de función
+2.0	Dirección inicial de registro	WORD	W#16#0040	Dirección inicial de registro
+4.0	Número de registros	INT	2	Número de registros

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#2123	Datos
+2.0	data[2]	WORD	W#16#2527	Datos

3.5.8 Código de función 04 – Read Input Registers

Finalidad y estructura

Función	Esta función permite leer registros individuales del esclavo.
Dirección inicial	El driver no comprueba el parámetro dirección inicial del registro y lo envía sin modificarlo.
Número de bits	Pueden leerse como máximo 110 registros (1 registro = 2 bytes).
LEN en bytes	6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#4	Código de función
+2.0	Dirección inicial de registro	WORD	W#16#0050	Dirección inicial de registro
+4.0	Número de registros	INT	3	Número de registros

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#2123	Datos
+2.0	data[2]	WORD	W#16#2527	Datos
+4.0	data[3]	WORD	W#16#3536	Datos

3.5.9 Código de función 05 -- Force Single Coil

Finalidad y estructura

Función	Con esta función es posible activar o desactivar un único bit en el esclavo.
Dirección de bit	El driver no comprueba el parámetro Dirección de bit y lo envía sin modificarlo.
Estado de bit	Como estado del bit se admiten los dos valores siguientes: FF00H => activar bit 0000H => desactivar bit
LEN en bytes	6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Clase	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#5	Código de función
+2.0	Dirección de bit	WORD	W#16#0019	Dirección de bit
+4.0	Estado de bit	WORD	W#16#FF00	Estado de bit

El esclavo tiene que devolver el telegrama de solicitud al maestro sin modificarlo (eco).

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Clase	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#5	Código de función
+2.0	Dirección de bit	WORD	W#16#0019	Dirección de bit
+4.0	Estado del bit	WORD	W#16#FF00	Estado del bit

3.5.10 Código de función 06 – Preset Single Register

Finalidad y estructura

Función	Con este comando puede sobrescribirse un registro de esclavo con un valor nuevo.
Dirección de registro	El driver no comprueba el parámetro dirección de registro y lo envía sin modificarlo.
Valor de registro	El valor de registro puede ser cualquier valor.
LEN en bytes	6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#6	Código de función
+2.0	Dirección de registro	WORD	W#16#0180	Dirección de registro
+4.0	Valor de registro	WORD	W#16#3E7F	Valor de registro

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#6	Código de función
+2.0	Dirección de registro	WORD	W#16#0180	Dirección de registro
+4.0	Valor de registro	WORD	W#16#3E7F	Valor de registro

3.5.11 Código de función 07 – Read Exception Status

Finalidad y estructura

Función Con este código de función pueden leerse 8 bits de evento del esclavo conectado. El número de bit inicial del bit de evento está definido por el equipo conectado y, por lo tanto, el programa de usuario SIMATIC no tiene que preajustarlo.

LEN en bytes 2

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#7	Código de función

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#3Exx	Datos

El driver registra los diferentes bits del telegrama de respuesta en el byte High del DB de destino data[1]. El byte Low de data[1] no se modifica. Como longitud del parámetro LEN se visualiza el valor 1. La longitud de recepción es siempre 1.

3.5.12 Código de función 08 -- Loop Back Diagnostic Test

Finalidad y estructura

Función	Esta función sirve para comprobar el enlace de comunicación. Para este código de función se soporta únicamente el código de diagnóstico 0000.
Código de diagnóstico	Para el parámetro Código de diagnóstico se admite únicamente el valor 0000.
Valor de test	Como valor de test se puede utilizar cualquier valor.
LEN en bytes	6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Clase	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#8	Código de función
+2.0	Código de diagnóstico	WORD	B#16#0000	Código de diagnóstico
+4.0	Valor de registro	WORD	B#16#A5C3	Valor de test

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Clase	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#8	Código de función
+2.0	Código de diagnóstico	WORD	B#16#0000	Código de diagnóstico
+4.0	Valor de test	WORD	B#16#A5C3	Valor de test

3.5.13 Código de función 11 – Fetch Communications Event Counter

Finalidad y estructura

Función	Este código de función permite leer del esclavo una palabra de estado de 2 bytes y un contador de eventos de 2 bytes.
LEN en bytes	2

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Clase	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#0B	Código de función

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Clase	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#FEDC	Palabra de estado
+2.0	data[2]	WORD	W#16#0108	Contador de eventos

3.5.14 Código de función 12 – Fetch Communications Event Log

Finalidad y estructura

Función Este código de función permite leer lo siguiente del esclavo:
-- Palabra de estado de 2 bytes
-- Contador de eventos de 2 bytes
-- Contador de telegramas de 2 bytes
-- Bytes de eventos de 64 bytes

LEN en bytes 2

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Clase	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#0C	Código de función

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Clase	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#8765	Palabra de estado
+2.0	data[2]	WORD	W#16#0108	Contador de eventos
+4.0	data[3]	WORD	W#16#0220	Contador de telegramas
+6.0	bytedata[1]	BYTE	B#16#01	Byte de eventos 1
+7.0	bytedata[2]	BYTE	B#16#12	Byte de eventos 2
:	:			:
+68.0	bytedata[63]	BYTE	B#16#C2	Byte de eventos 63
+69.0	bytedata[64]	BYTE	B#16#D3	Byte de eventos 64

3.5.15 Código de función 15 – Force Multiple Coils

Finalidad y estructura

Función	Con este código de función pueden modificarse como máximo 1696 bits en el esclavo.
Dirección inicial	El driver no comprueba el parámetro dirección inicial de bit y lo envía sin modificarlo.
Número de bits	Como número de bits (number of coils) se admite cualquier valor entre 1 y 1696. Con ello se define cuántos bits pueden sobrescribirse en el esclavo. El driver crea el parámetro "contador de bytes" incluido en el telegrama de solicitud a causa del parámetro "número de bits" transferido.
LEN en bytes	> 6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#0F	Código de función
+2.0	Dirección inicial de bit	WORD	W#16#0058	Dirección inicial de bit
+4.0	Número de bits	INT	10	Número de bits
+6.0	coil_state[1]	WORD	W#16#EFCD	Status Coil 5FH..58H/57H..50H

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#F	Código de función
+2.0	Dirección de bit	WORD	W#16#0058	Dirección de bit
+4.0	Número de bits	INT	10	Número de bits

El driver envía los datos del DB de origen/destino por palabras. El byte High (byte 1) de la dirección de palabra "EF" en el DB se envía en primer lugar, seguido del byte Low (byte 0) de la dirección de palabra "CD" del DB. Si se envía un número impar de bytes, el último byte es el byte High (byte 1).

3.5.16 Código de función 16 – Preset Multiple Registers

Finalidad y estructura

Función	El código de función 16 permite sobrescribir un máximo de 109 registros en el esclavo con un telegrama de solicitud.
Dirección inicial	El driver no comprueba el parámetro dirección inicial del registro y lo envía sin modificarlo.
Número de registros	Pueden leerse como máximo 109 registros (1 registro = 2 bytes). El driver crea el parámetro "contador de bytes" incluido en el telegrama de solicitud a causa del parámetro "número de registros" transferido.
LEN en bytes	> 6

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#10	Código de función
+2.0	Dirección inicial de registro	WORD	W#16#0060	Dirección inicial de registro
+4.0	Número de registros	INT	3	Número de registros
+6.0	reg_data[1]	WORD	W#16#41A1	Datos de registro
+8.0	reg_data[2]	WORD	W#16#42A2	Datos de registro
+10.0	reg_data[3]	WORD	W#16#43A3	Datos de registro

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#10	Código de función
+2.0	Dirección inicial de registro	WORD	W#16#0060	Dirección inicial de registro
+4.0	Número de registros	INT	3	Número de registros

3.6 Driver esclavo de Modbus

3.6.1 Componentes del acoplamiento de esclavo Modbus

Introducción

Junto con el bloque de función correspondiente, este driver permite configurar un enlace de comunicación entre un sistema de automatización maestro Modbus y el módulo de comunicación esclavo Modbus ET 200S en forma de sistema apto para Modbus.

Principio de la transmisión de datos

Para la transferencia, el protocolo Modbus se utiliza en formato RTU. La transferencia de datos se desarrolla siguiendo el principio maestroesclavo. El maestro se inicializa durante la transferencia, de modo que el módulo y la CPU S7 se utilizan como esclavos. Los códigos de función 01, 02, 03, 04, 05, 06, 08, 15 y 16 pueden emplearse para la comunicación entre el módulo y el sistema maestro. El driver evalúa la dirección Modbus del telegrama de solicitud del maestro como si se tratase de una S7. Ello significa que pueden leerse las siguientes áreas de la CPU S7:

- Leer y escribir marcas, salidas, bloques de datos
- Leer marcas, entradas, temporizadores, contadores

Con el acoplamiento existente, los accesos a datos del protocolo MODBUS se convierten a las áreas de memoria específicas de la CPU SIMATIC S7.

Estructura de datos

Antes de configurar los datos S7 del proyecto hay que asegurarse de que los datos son compatibles con los programas de usuario del sistema maestro Modbus.

Acoplamiento de esclavo Modbus

El acoplamiento de esclavo Modbus para el módulo está formado por dos partes:

- Driver esclavo Modbus
- Bloque de función comunicación Modbus para la CPU SIMATIC S7

FB de comunicación esclavo Modbus

Para el acoplamiento de esclavo Modbus se requiere un FB de comunicación especial en la CPU S7, además del driver esclavo Modbus.

El FB de comunicación Modbus procesa todas las funciones necesarias para el acoplamiento.

El FB81(S_MODB) recibe el protocolo Modbus y convierte las direcciones de Modbus en áreas de memoria SIMATIC.

En el programa de usuario, el FB81 debe llamarse en el programa cíclico. El FB de comunicación Modbus emplea un bloque de datos de instancia como zona de trabajo.

3.6.2 Transferencia de datos con el esclavo Modbus ET 200S

Fases de la transmisión de datos

Para ejecutar una petición de esclavo Modbus debe activarse cíclicamente el FB S_MODB en el programa de usuario. S_MODB recibe la petición del módulo interfaz serie ET 200S Modbus/USS, la ejecuta y devuelve la respuesta al módulo. La comunicación entre la CPU y el módulo se realiza mediante los bloques de función S_SEND y S_RCV, que se llaman con S_MODB.

Tras cada rearranque de la CPU, el programa de usuario tiene que inicializar el FB de comunicación Modbus. La inicialización se activa con un flanco positivo en la entrada CP_START. El FB registra los tamaños de las áreas de operandos E, A, M, T y Z de la CPU en el bloque de datos de instancia del FB. Una vez finalizada correctamente la inicialización, el FB activa la salida CP_START_OK.

La salida CP_START_ERROR indica un error de inicialización. En este caso, no es posible una comunicación Modbus y todas las peticiones del maestro Modbus se responden con un mensaje de código de excepción.

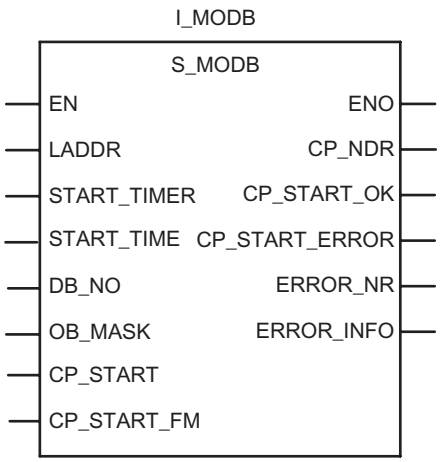
S_MODB utiliza una tabla de conversión de datos Modbus, que se encuentra en el bloque de datos, para reproducir las direcciones de Modbus en las áreas de memoria de SIMATIC S7.

El parámetro de entrada OB_MASK permite indicar al FB Modbus que enmascare errores de acceso E/S. Si se produce un acceso de escritura a periferia no existente, la CPU no pasa al estado operativo STOP y no llama ningún OB de error. El FB detecta el error de acceso y la función se finaliza con una respuesta de error en el maestro Modbus.

Representación AWL

```
CALL                    S_MODB, I_MODB
       LADDR             =
       START_TIMER       =
       START_TIME        =
       DB_NO             =
       OB_MASK           =
       CP_START          =
       CP_START_FM       =
       CP_NDR            =
       CP_START_OK       =
       CP_START_ERROR    =
       ERROR_NR          =
       ERROR_INFO        =
```

Representación KOP



Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

3.6.3 Áreas de datos en la CPU SIMATIC

Tabla de conversión de datos Modbus

El FB81(S_MODB) interpreta las direcciones de Modbus incluidas en los telegramas de acuerdo con S7 y las convierte al área de memoria SIMATIC. El usuario puede indicar el acceso a las diferentes áreas de memoria SIMATIC transfiriendo un DB como entrada para FB81(S_MODB) (véase la tabla).

Tabla 3- 17 Tabla de conversión

Dirección	Nombre	Tipo	Valor inicial	Valor actual	Comentario	Código de función aplicable
0.0	aaaaa	WORD	W#16#0	W#16#0	Principio de dirección de Modbus	01, 05, 15
2.0	bbbbb	WORD	W#16#0	W#16#7F7	Fin de dirección de Modbus	
4.0	uuuuu	WORD	W#16#0	W#16#1F4	Marcas	
6.0	ccccc	WORD	W#16#0	W#16#7F8	Principio de dirección de Modbus	01, 05, 15
8.0	ddddd	WORD	W#16#0	W#16#FEF	Fin de dirección de Modbus	
10.0	ooooo	WORD	W#16#0	W#16#15	Salidas	
12.0	eeeee	WORD	W#16#0	W#16#FF0	Principio de dirección de Modbus	01, 05, 15
14.0	fffff	WORD	W#16#0	W#16#17E7	Fin de dirección de Modbus	
16.0	ttttt	WORD	W#16#0	W#16#28	Temporizadores	
18.0	ggggg	WORD	W#16#0	W#16#17E8	Principio de dirección de Modbus	01, 05, 15
20.0	hhhhh	WORD	W#16#0	W#16#1FDF	Fin de dirección de Modbus	
22.0	zzzzz	WORD	W#16#0	W#16#28	Contadores	
24.0	kkkkk	WORD	W#16#0	W#16#1FE0	Principio de dirección de Modbus	02
26.0	lllll	WORD	W#16#0	W#16#27D7	Fin de dirección de Modbus	02
28.0	vvvvv	WORD	W#16#0	W#16#320	Marcas	02
30.0	nnnnn	WORD	W#16#0	W#16#27D8	Principio de dirección de Modbus	02
32.0	rrrrr	WORD	W#16#0	W#16#2FCF	Fin de dirección de Modbus	02

Dirección	Nombre	Tipo	Valor inicial	Valor actual	Comentario	Código de función aplicable
34.0	sssss	WORD	W#16#0	W#16#11	Entradas	02
36.0	DB_Number_FC_03_06_16	WORD	W#16#0	W#16#6	DB	03, 06, 15
38.0	DB_Number_FC_04	WORD	W#16#0	W#16#2	DB	04
40.0	DB_Min	WORD	W#16#0	W#16#1	Número de DB más pequeño empleado	Límites
42.0	DB_Max	WORD	W#16#0	W#16#6	Número de DB más grande empleado	Límites
44.0	M_Min	WORD	W#16#0	W#16#1F4	Marca más pequeña empleada	Límites
46.0	M_Max	WORD	W#16#0	W#16#4B0	Marca más grande empleada	Límites
48.0	Q_Min	WORD	W#16#0	W#16#0	Salida más pequeña empleada	Límites
50.0	Q_Max	WORD	W#16#0	W#16#64	Salida más grande empleada	Límites

3.6.4 Configurar los parámetros para el acoplamiento

Parámetros de la configuración de hardware

Los siguientes parámetros y estados operativos tienen que ajustarse en la configuración de hardware para el driver.

- Velocidad de transferencia, paridad
- Dirección de esclavo del módulo
- Estado operativo (normal, supresión de frecuencias perturbadoras)
- Factor de multiplicación para el tiempo de retardo entre caracteres

Parámetros del DB de entrada para el FB81

Los parámetros que aparecen a continuación deben ajustarse con el DB de entrada para el FB81(S_MODB).

- Áreas de direccionamiento para los códigos de función 01, 05, 15
- Áreas de direccionamiento para el código de función 02
- Número de DB básico para los códigos de función 03, 06, 16
- Número de DB básico para el código de función 04
- Límites para accesos de escritura

Parametrización del driver esclavo

La tabla lista los parámetros que pueden ajustarse para el driver Modbus del módulo.

Tabla 3- 18 Parámetros del driver esclavo Modbus

Parámetro	Descripción	Rango de valores	Valor por defecto
Alarma de diagnóstico	Indique si el módulo crea una alarma de diagnóstico cuando surge un error grave.	• No • Sí	No
Activar reconocimiento BREAK	Si se produce una ruptura de la línea o no se ha conectado ningún cable de interfaz, el módulo notifica el mensaje de error "Break".	• No • Sí	No
Tipo de interfaz	Indique la interfaz eléctrica que debe emplearse.	• RS-232C • RS-422 (dúplex) • RS-485 (semidúplex)	RS-232C

Parámetro	Descripción	Rango de valores	Valor por defecto
Preajuste dúplex y semidúplex de la línea de recepción	Indique el preajuste de la línea de recepción en los modos de operación RS-422 y RS-485. No en el modo de operación RS232C. La configuración "Nivel invertido" sólo es requerida en caso de repuesto para asegurar la compatibilidad.	RS422: R(A) 5V / R(B) 0V (BREAK) R(A) 0V / R(B) 5V Nivel invertido RS485: sin R(A) 0V / R(B) 5V	RS422: R(A) 5V / R(B) 0V (BREAK) RS485: R(A) 0V / R(B) 5V
Control de flujo de datos (con parámetros predeterminados; cambiar valores predeterminados en el programa de usuario)	Se pueden enviar y recibir datos con control del flujo de datos. Mediante el control del flujo de datos, la transmisión de datos se sincroniza cuando un interlocutor trabaja más rápidamente que el otro. Seleccione el tipo de control del flujo de datos y ajuste los parámetros correspondientes. Nota: Con la interfaz RS 485 no es posible el control de flujo de datos. El control del flujo de datos con "Manejo automático de las señales V24" sólo es posible con la interfaz RS232C.	• Sin • Manejo automático de las señales V.24	Sin
Velocidad de transmisión	Seleccione la velocidad de la transmisión de datos en bits por segundo.	• 110 • 300 • 600 • 1.200 • 2.400 • 4.800 • 9.600 • 19.200 • 38.400 • 57.600 • 76.800 • 115.200	9600
Bits de parada	Seleccione la cantidad de bits de parada que se colocan detrás de cada carácter durante la transferencia y que marcan el fin de un carácter.	• 1 • 2	1

Parámetro	Descripción	Rango de valores	Valor por defecto
Paridad	La secuencia de los bits de datos puede ampliarse en un carácter para incluir el bit de paridad. El valor adicional (0 ó 1) traslada el valor de todos los bits (bits de datos y bit de paridad) a un estado definido. **Sin**: Los datos se envían sin bit de paridad. **Impar**: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es impar con el estado de señal "1". **Par**: El bit de paridad se activa, de modo que el número total de bits de datos (inclusive el bit de paridad) es par con el estado de señal "1".	• Sin • Impar • Par	Par
Dirección del esclavo	Dirección de esclavo propia del módulo	1-247	222
Modo de operación	• Modo normal • Supresión de frecuencias perturbadoras	• Normal • Supresión de frecuencias perturbadoras	Normal
Multiplicador de retardo de caracteres	Utiliza un multiplicador del tiempo de retardo entre caracteres de 1-10.	1 a 10	1
Borrar búfer de recepción de la interfaz serie al arrancar	Indique si el búfer de recepción del módulo interfaz serie debe borrarse automáticamente cuando la CPU pasa del estado operativo STOP a RUN (arranque de la CPU). Ello permite asegurarse de que el búfer de recepción de la interfaz serie sólo recibe telegramas recibidos después de arrancar la CPU.	• No • Sí	Sí
[1] El tiempo de retardo entre caracteres más breve se calcula según la velocidad de transmisión.			

En la siguiente lista encontrará explicaciones sobre los diferentes parámetros o valores:

- **Dúplex (RS422), modo a cuatro hilos**

  En este modo de operación se envían datos a través de la línea de envío T(A), T(B) y se reciben a través de la línea de recepción R(A), R(B). El tratamiento de errores se produce según la función configurada con el parámetro “Modo de operación del driver" (normal o supresión de frecuencias perturbadoras).

- **Semidúplex (RS485), modo a dos hilos**

  En este modo de operación, el driver ejecuta un modo de conmutación de la línea de recepción de 2 hilos R(A), R(B) de la interfaz entre modo de envío y de recepción. El inicio de un telegrama de recepción es reconocido a través de la dirección correcta de esclavo. Al utilizar el acoplamiento punto y punto, se recomienda preajustar la preasignación de la línea de recepción con los parámetros
  R (A) 0V, R(B) 5V.

- **Ocupación estándar de la línea de recepción**

  Este parámetro indica el estado base de la línea de recepción en los modos de operación RS-422 y RS-485. No se utiliza para el modo de operación RS232C.

  – **R(A) 5V, R(B) 0V (BREAK)**
    El módulo preajusta la siguiente asignación para la línea de dos hilos R(A), R(B):
    R(A) --> +5V, R(B) --> 0V ($V_A - V_B \geq +0{,}3$ V).
    Esto significa que en el módulo hay nivel BREAK con rotura de hilo.

  – **R(A) 0V, R(B) 5V (High)**
    El módulo preajusta la siguiente asignación para la línea de dos hilos R(A),R(B):
    R(A) --> 0V, R(B) --> +5 V ($V_A - V_B \leq -0{,}3$ V).
    Esto significa que en el módulo hay nivel HIGH con rotura de línea (o bien en estado de reposo si ningún esclavo envía). El estado de línea BREAK no puede detectarse.

  – **Sin** (sólo con RS485)
    En una conexión multipunto, la preasignación de la línea de recepción está desactivada.

- **Velocidad de transmisión**

  La velocidad de transmisión es la velocidad de transferencia de datos en bits por segundo (bps). La velocidad de transmisión del módulo es de 38400 bps en modo semidúplex.

- **Bits de datos**

  La cantidad de bits de datos describe en cuántos bits puede representarse un carácter que debe transferirse. Para este driver deben ajustarse siempre 8 bits de datos. Hay que emplear siempre una trama de caracteres de 11 bits. Si se ajusta paridad "sin", deben seleccionarse 2 bits de parada.

- **Bits de parada**

  La cantidad de bits de parada define la distancia temporal mínima entre dos caracteres que deben transferirse. Hay que emplear siempre una trama de caracteres de 11 bits. Si se ajusta paridad "sin", deben seleccionarse 2 bits de parada.

- **Paridad**

  El bit de paridad se utiliza para la seguridad de los datos. Complementa la cantidad de bits de datos transferidos hasta obtener un número par o impar en función de la parametrización. Si se ajusta la paridad "sin" no se transfiere ningún bit de paridad. Esto reduce la seguridad de transmisión. Hay que emplear siempre una trama de caracteres de 11 bits. Si se ajusta paridad "sin", deben seleccionarse 2 bits de parada.

- **Dirección del esclavo**

  Aquí se indica la dirección propia de esclavo Modbus a la que debe responder el módulo. El módulo sólo responde telegramas en los que la dirección de esclavo recibida sea idéntica a la dirección de esclavo propia parametrizada. Los telegramas a otros esclavos no se comprueban ni se responden.

- **Modo normal**

  En este modo de operación, todos los errores de transmisión o BREAK detectados antes y después del telegrama de recepción del esclavo provocan un mensaje de error pertinente.

- **Supresión de frecuencias perturbadoras**

  Si al principio del telegrama de recepción se reconoce BREAK en la línea de recepción o si el bloque de interfaz del módulo determina errores de transmisión, el driver ignora esta recepción defectuosa. El inicio de un telegrama de recepción es reconocido a través de la dirección correcta de esclavo. También se ignoran errores de transmisión o BREAK que aparecen una vez finalizado el telegrama de recepción (código CRC).

- **Multiplicador de retardo de caracteres**

  Cuando un interlocutor no puede cumplir las exigencias en cuanto a tiempo de la especificación Modbus, existe la posibilidad de multiplicar el tiempo de retardo entre caracteres t_{TRC} por el factor de multiplicación f_{MUL}. El tiempo de retardo entre caracteres sólo debe modificarse en aquellos casos en que el interlocutor del acoplamiento no puede mantener el tiempo requerido.

  El tiempo de retardo entre caracteres modificado t_{TRC} resulta de:

$t_{TRC} = t_{ZVZ\_TAB} * f_{MUL}$

t_{TRC_TAB} = Valor de tabla para t_{TRC}
f_{MUL} = Factor de multiplicación

Nota

Tenga en cuenta también las indicaciones recogidas en los temas Datos de identificación (Página 63) y Cargar actualizaciones de firmware a posteriori (Página 65).

3.6.5 Códigos de función del esclavo

Códigos de función del driver esclavo Modbus

El driver esclavo Modbus soporta los códigos de función listados en la tabla .

Nota

Todas las direcciones de Modbus listadas en la tabla siguiente hacen referencia al nivel de telegrama de transferencia y no al nivel de usuario del sistema maestro Modbus. Esto significa que las direcciones de Modbus empiezan por 0000 Hex en los telegramas de transferencia.

Tabla 3- 19 Códigos de función de esclavo

Código de función	Descripción	Función en SIMATIC S7	
01	Read Coil Status	Leer por bits	Marca M
		Leer por bits	Salidas A
		Leer por bits (intervalo de 16 bits)	Temporizadores T
		Leer por bits (intervalo de 16 bits)	Contadores Z
02	Read Input Status	Leer por bits	Marca M
		Leer por bits	Entradas E
03	Read Holding Registers	Leer por palabras	Bloque de datos DB
04	Read Input Registers	Leer por palabras	Bloque de datos DB
05	Force Single Coil	Escribir por bits	Marca M
		Escribir por bits	Salidas A
06	Preset Single Register	Escribir por palabras	Bloque de datos DB
08	Loop Back Test	-	-
15	Force Multiple Coils	Escribir por bits (1...2040 bits)	Marca M
		Escribir por bits (1...2040 bits)	Salidas A
16	Preset multiple (holding) registers	Escribir por palabras (1...127 registros)	Bloque de datos DB

3.6.6 Código de función 01 – Read Coil (Output) Status

Finalidad y estructura

El código de función 01 - Read Coil (Output) Status presenta las siguientes características:

Función	Esta función permite leer diferentes bits de las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro Modbus.				
Telegrama de solicitud	ADDR	FUNC	start_adress	bit_number	CRC
Telegrama de respuesta	ADDR	FUNC	start_adress	n Byte DATA	CRC
LEN en bytes	6				

start_address

La dirección de bit de Modbus "start_address" es interpretada por el driver. Ejemplo: El FB81(S_MODB) comprueba si "start_address" se encuentra en una de las áreas que se indicaron en el DB de conversión para FC 01, 05, 15 (de/a: marcas, salidas, temporizadores, contadores).

Si la dirección de bits de Modbusstart_address se encuentra en el área	**se accede al área de memoria SIMATIC**	
De *aaaaa* a *bbbbb*	A partir de la marca	M *uuuuu*.0
De *ccccc* a *ddddd*	A partir de la salida	A *ooooo*.0
De *eeeee* a *fffff*	A partir del temporizador	T *ttttt*
De *ggggg* a *hhhhh*	A partir del contador	Z *zzzzz*

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza del modo siguiente:

Acceso empezando con SIMATIC	**Fórmula de conversión**	
Byte de marca	=((start_address- *aaaaa*) / 8)	+ *uuuuu*
Byte de salida	=((start_address- *ccccc*) / 8)	+ *ooooo*
Temporizador	=((start_address- *eeeee*) / 16)	+ *ttttt*
Contadores	=((start_address- *ggggg*) / 16)	+ *zzzzz*

Acceso a marcas y salidas

Al acceder a las áreas SIMATIC de marcas y salidas, se calcula el número de bit restante y se utiliza para direccionar el bit en cuestión dentro del primer/último byte de marca o de salida.

Acceso a temporizadores y contadores.

Al calcular las direcciones, el resultado

- (start_address - eeeee) o
- (start_address – ggggg)

debe ser divisible por 16 sin resto (sólo acceso por palabras empezando en el límite de palabra).

bit_number

El bit_number (Number of coils) puede ser cualquier valor entre 1 y 1768. Se lee esta cantidad de bits.

Al acceder a las áreas SIMATIC de temporizadores y contadores, "bit_number" tiene que ser divisible por 16 (sólo acceso por palabras).

Ejemplo de aplicación

Tabla 3- 20 Ejemplo de conversión del direccionamiento Modbus:

Conversión del direccionamiento Modbus para los códigos de función FC 01, 05 y 15	
Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC
De *0* a *2047*	A partir de la marca M *1000*.0
De *2048* a *2559*	A partir de la salida A *256*.0
De *4096* a *4607*	A partir del temporizador T *100*
De *4608* a *5119*	A partir del contador Z *200*

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#1	Código de función
+2.0	Dirección inicial de bit	WORD	W#16#0040	Dirección inicial de bit
+4.0	Número de bits	INT	16	Número de bits

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#1701	Datos

El driver registra los datos del telegrama de respuesta en el DB de destino por palabras. El primer byte recibido se registra como byte Low de la primera palabra "data[1]", el tercer byte recibido como byte Low de la segunda palabra "data[2]", etc. Si se leen menos de 9 bits o si sólo se ha leído un byte Low, en el byte High restante de la última palabra se registra el valor 00H.

Cálculo de direcciones:

La dirección Modbus "start_address" 0040 Hex (64 decimal) se encuentra en el área de marcas:

Byte de marca	=((start_address- *aaaaa*)	/ 8)	+ *uuuuu*
	=((64 - *0*)	/ 8)	+ *1000*
	=1008;		

El número de bit restante da:

Nº bit restante	=((start_address- *aaaaa*)	% 8)	[Módulo 8]
	=((64 - *0*)	% 8)	
	= 0;		

Se produce un acceso que empieza con la marca M 1008.0 hasta M 1011.7 inclusive.

Número de bits:

El número de bits de Modbus “bit_number" 0020 Hex (32 decimal) significa que deben leerse 32 bits = 4 bytes.

La tabla lista más ejemplos del acceso a datos.

Tabla 3- 21 Otros ejemplos del acceso a datos

start_address Hex	decimal	Cálculo de direcciones	Dirección
0000	0	Marca ((0 – *0*) / 8) + *1000*	-> M1000.0
0021	33	Marca ((33 – *0)* / 8) + *1000*	-> M1004.1
0400	1024	Marca ((1024 – *0*) / 8) + *1000*	-> M1128.0
0606	1542	Marca ((1542 – *0*) / 8) + *1000*	-> M1192.6
0840	2112	Salida ((2112 – *2048*) / 8) + *256*	-> A264.0
09E4	2532	Salida ((2532 – *2048*) / 8) + *256*	-> A316.4
1010	4112	Temporizadores ((4112 – *4096*) / 16) + *100*	-> T 101
10C0	4288	Temporizadores ((4288 – *4096*) / 16) + *100*	-> T112
1200	4608	Contadores ((4608 – *4608*) / 16) + *200*	-> Z 200
13E0	5088	Contadores ((5088 – *4608*) / 16) + *200*	-> Z 230

3.6.7 Código de función 02 – Read Input Status

Finalidad y estructura

El código de función 02 – Read Input Status presenta las siguientes características:

Función	Esta función permite leer diferentes bits de las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro Modbus.				
Telegrama de solicitud	ADDR	FUNC	start_adress	bit_number	CRC
Telegrama de respuesta	ADDR	FUNC	Byte_count n	n Byte DATA	CRC
LEN en bytes	6				

start_address

La dirección de bit Modbus "start_address" es interpretada por el driver del modo siguiente:

El driver comprueba si "start_address" se encuentra dentro de una de las áreas que se entraron en el DB de conversión para el FC 02 (de/a: marcas, entradas).

Si la dirección de bits de Modbusstart_address se encuentra en el área	**se accede al área de memoria SIMATIC**	
De *kkkkk* a *lllll*	A partir de la marca	M *vvvvv*.0
De *nnnnn* a *rrrrr*	A partir de la entrada	E *sssss*. 0

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza del modo siguiente:

Acceso empezando con SIMATIC	**Fórmula de conversión**	
Byte de marca	= ((start_address- *kkkkk*) / 8)	+ *vvvvv*
Byte de entrada	= ((start_address- *nnnnn*) / 8)	+ *sssss*

Acceso a marcas y entradas

Al acceder a las áreas SIMATIC de marcas y entradas, se calcula el número de bit restante y se utiliza para direccionar el bit en cuestión dentro del primer/último byte de marca o de entrada.

bit_number

Como bit_number (Number of coils) se admite cualquier valor entre 1 y 1768. Se lee esta cantidad de bits.

Ejemplo de aplicación

Ejemplo de conversión de la asignación de direcciones Modbus:

Tabla 3- 22 Conversión del direccionamiento Modbus para el código de función FC 02

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC	
De *0* a *4095*	A partir de la marca	M 2*000.0*
De *4096* a *5119*	A partir de la entrada 0	I *128.0*

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#2	Código de función
+2.0	Dirección inicial de bit	WORD	W#16#0120	Dirección inicial de bit
+4.0	Número de bits	INT	24	Número de bits

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Data[1]	WORD	W#16#2604	Datos
+2.0	Data[2]	WORD	W#16#0048	Datos

El driver registra los datos del telegrama de respuesta en el DB de destino por palabras. El primer byte recibido se registra como byte Low de la primera palabra "data[1]", el tercer byte recibido como byte Low de la segunda palabra "data[2]", etc.

Si se leen menos de 9 bits o si sólo se ha leído un byte Low, en el byte High restante de la última palabra se registra el valor 00H.

Cálculo de direcciones:

La dirección Modbus "start_address" 1030 Hex (4144 decimal) se encuentra en el área de entradas:

Byte de entrada =((start_address- *nnnnn*) / 8) + *sssss*
=((4144 - *4096*) / 8) + *128*
=134;

El número de bit restante da:

Nº bit restante =((start_address- *aaaaa*) % 8) [Módulo 8]
=((4144 - *4096*) % 8)
= 0;

Se produce un acceso que empieza con la entrada E 134.0 hasta E 136.7 inclusive.

Número de bits:

El número de bits de Modbus “bit_number" 0018 Hex (24 decimal) significa que deben leerse 24 bits = 3 bytes.

La tabla lista más ejemplos del acceso a datos.

Tabla 3- 23 Otros ejemplos del acceso a datos

start_address Hex	decimal	Cálculo de direcciones	Dirección
0000	0	Marca ((0 -*0*) / 8) + *2000*	-> M2000.0
0071	113	Marca ((113 -*0*) / 8) + *2000*	-> M2014.1
0800	2048	Marca ((2048 -*0*) / 8) + *2000*	-> M2256.0
0D05	3333	Marca ((3333 -*0*) / 8) + *2000*	-> M2416.5
1000	4096	Entrada ((4096 -*4096*) / 8) + *128*	-> E 128.0
10A4	4260	Entrada ((4260 -*4096*) / 8) + *128*	-> E 148.4

3.6.8 Código de función 03 – Read Output Registers

Finalidad y estructura

El código de función 03 - Read Output Registers presenta las siguientes características:

Función	Esta función permite leer palabras de datos de un bloque de datos a través del sistema maestro Modbus.				
Telegrama de solicitud	ADDR	FUNC	start_adress	register_number	CRC
Telegrama de respuesta	ADDR	FUNC	Byte_count n	n/2-Register DATA (High, Low)	CRC
LEN en bytes	6				

start_address

La dirección de registro Modbus "start_register" es interpretada por el driver del modo siguiente:

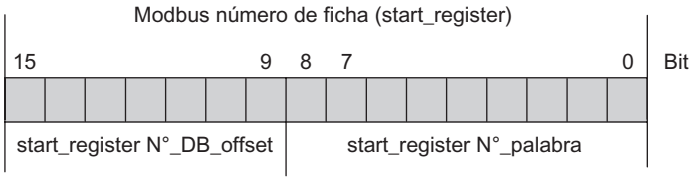


Figura 3-10 Interpretación del número de registro Modbus

Para seguir creando direcciones, el FB81(S_MODB) utiliza el número de DB básico (de DB xxxxx) registrado en el DB de conversión para los FC 03, 06 y 16.

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza en dos pasos:

Acceso a SIMATIC	Fórmula de conversión
DB de bloque de datos (DB resultante)	=(Número de DB básico *xxxxx*+start_register Nº_DB_Offset)
Palabra de datos DBW	=(start_register Nº_palabra *2)

Fórmula de cálculo para start_register

Si se conoce el DB resultante que debe leerse, la dirección de Modbus start_register que se requiere en el sistema maestro puede calcularse según la fórmula siguiente:

start_register = ((DB resultante - número de DB básico) * 512) + (palabra de datos_DBW / 2)

Para ello sólo deben tomarse como base números pares de palabras de datos.

register_number

Como register_number (Number of registers) se admite cualquier valor entre 1 y 110. Se lee esta cantidad de registros.

Ejemplo de aplicación

Tabla 3- 24 Conversión del direccionamiento Modbus para los códigos de función FC 03, 06 y 16

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC
0	A partir del bloque de datos DB *800* (número de DB básico)

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#3	Código de función
+2.0	Dirección inicial de registro	WORD	W#16#0040	Dirección inicial de registro
+4.0	Número de registros	INT	2	Número de registros

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Data[1]	WORD	W#16#2123	Datos
+2.0	Data[2]	WORD	W#16#2527	Datos

Cálculo de direcciones:

La dirección Modbus "start_register" 0050 Hex (80 decimal) se interpreta del modo siguiente:

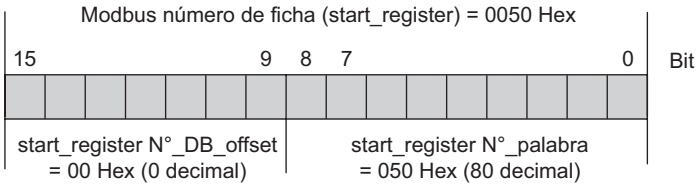


Figura 3-11 Interpretación del número de registro Modbus 0050 Hex

DB de bloque de datos (DB resultante) = (Número de DB básico *xxxxx*+start_register - Nº_DB_Offset)
= (*800* + 0)
= 800 ;

Palabra de datos DBW =(start_register Nº_palabra *2)
= (80 * 2)
= 60 ;

Se accede al DB 800, palabra de datos DBW 160.

Número de registros:

El número de registros Modbus "register_number" 0002 Hex (2 decimal) significa que deben leerse 2 registros = 2 palabras de datos.

La tabla lista más ejemplos del acceso a datos.

Tabla 3- 25 Otros ejemplos del acceso a datos

start_register							
start_register		Nº_DB_básico	Nº_DB_offset	Número de palabra		DB resultante	DBW
Hex	Decimal	Decimal	Decimal	Hex	Decimal	Decimal	Decimal
0000	0	800	0	000	0	800	0
01F4	500	800	0	1F4	500	800	1000
0200	512	800	1	000	0	801	0
02FF	767	800	1	0FF	255	801	510
0300	768	800	1	100	256	801	512
03FF	1023	800	1	1FF	511	801	1022
0400	1024	800	2	000	0	802	0

3.6.9 Código de función 04 – Read Input Registers

Finalidad y estructura

El código de función 04 - Read Input Registers presenta las siguientes características:

Función	Esta función permite leer palabras de datos de un bloque de datos a través del sistema maestro Modbus.				
Telegrama de solicitud	ADDR	FUNC	start_register	register_number	CRC
Telegrama de respuesta	ADDR	FUNC	Byte_count n	n/2-Register DATA (High, Low)	CRC
LEN en bytes	6				

start_address

La dirección de registro Modbus "start_register" es interpretada por el driver del modo siguiente:

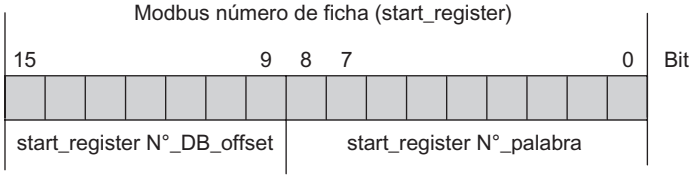


Figura 3-12 Interpretación del número de registro Modbus

Para seguir creando direcciones, el FB81(S_MODB) utiliza el número de DB básico (de DB xxxxx) registrado en el DB de conversión para el FC 04.

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza en dos pasos:

Acceso a SIMATIC	Fórmula de conversión
DB de bloque de datos (DB resultante)	=(Número de DB básico *xxxxx*+start_register Nº_DB_Offset)
Palabra de datos DBW	=(start_register Nº_palabra *2)

Fórmula de cálculo para start_register

Si se conoce el DB resultante que debe leerse, la dirección de Modbus start_register que se requiere en el sistema maestro puede calcularse según la fórmula siguiente:

start_register = ((DB resultante - número de DB básico) * 512) + (palabra de datos_DBW / 2)

Para ello sólo deben tomarse como base números pares de palabras de datos.

register_number

Como register_number (Number of registers) se admite cualquier valor entre 1 y 110. Se lee esta cantidad de registros.

Ejemplo de aplicación

Tabla 3- 26 Conversión del direccionamiento Modbus para el código de función FC 04

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC
0	A partir del bloque de datos DB *900*(número de DB básico)

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#4	Código de función
+2.0	Dirección inicial de registro	WORD	W#16#0050	Dirección inicial de registro
+4.0	Número de registros	INT	3	Número de registros

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Data[1]	WORD	W#16#2123	Datos
+2.0	Data[2]	WORD	W#16#2527	Datos
+4.0	Data[3]	WORD	W#16#3536	Datos

Cálculo de direcciones:

La dirección Modbus "start_register" 02C0 Hex (704 decimal) se interpreta del modo siguiente:

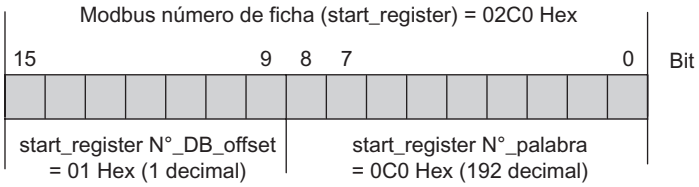


Figura 3-13 Interpretación del número de registro Modbus 0270 Hex

DB de bloque de datos (DB resultante) = (Número de DB básico *xxxxx*+start_register - Nº_DB_Offset)
= (900+ 0)
= 901;

Palabra de datos DBW =(start_register Nº_palabra *2)
= (192 * 2)
= 384;

Se accede al DB 901, palabra de datos DBW 384.

Número de registros:

El número de registros Modbus "register_number" 0003 Hex (3 decimal) significa que deben leerse 3 registros = 3 palabras de datos.

La tabla lista más ejemplos del acceso a datos.

Tabla 3- 27 Otros ejemplos del acceso a datos

start_register							
start_register		Nº_DB_básico	Nº_DB_offset	Número de palabra		DB resultante	DBW
Hex	Decimal	Decimal	Decimal	Hex	Decimal	Decimal	Decimal
0000	0	900	0	000	0	900	0
0064	100	900	0	064	100	900	200
00C8	200	900	0	0C8	200	900	400
0190	400	900	0	190	400	900	800
1400	5120	900	10	000	0	910	0
1464	5220	900	10	064	100	910	200
14C8	5320	900	10	0C8	200	910	400

3.6.10 Código de función 05 – Force Single Coil

Finalidad y estructura

El código de función 05 – Force Single Coil presenta las siguientes características:

Función	Esta función permite escribir un bit en las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro Modbus.				
Telegrama de solicitud	ADDR	FUNC	coil_address	DATA-on/off	CRC
Telegrama de respuesta	ADDR	FUNC	coil_address	DATA-on/off	CRC
LEN en bytes	6				

coil_address

La dirección de bit Modbus "coil_address" es interpretada por el driver del modo siguiente:

FB81(S_MODB) comprueba si "coil_address" se encuentra en una de las áreas que se indicaron en el DB de conversión para los FC 01, 05 y 15 (de/a: marcas, salidas, temporizadores, contadores).

Si la dirección de bits de Modbusstart_address se encuentra en el área	**se accede al área de memoria SIMATIC**	
De *aaaaa* a *bbbbb*	A partir de la marca	M *uuuu*.0
De *ccccc* a *ddddd*	A partir de la salida	A *oooo*.0

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza en dos pasos:

Acceso empezando con SIMATIC	**Fórmula de conversión**	
Byte de marca	=((start_address- *cccc*) / 8)	+ *ooooo*
Byte de salida	=((start_address *aaaa*) / 8)	+ *uuuuu*

Acceso a marcas y salidas

Al acceder a las áreas SIMATIC de marcas y salidas, se calcula el número de bit restante y se utiliza para direccionar el bit en cuestión dentro del byte de marca o de salida.

Acceso a temporizadores y contadores.

Con este código de función FC 05 no se permite acceder a las áreas SIMATIC de temporizadores y contadores y el driver rechaza los intentos de acceso con un telegrama de error.

DATA-on/off

DATAon/off admite los dos valores siguientes:

FF00H = activar bit.

0000H = borrar bit.

Ejemplo de aplicación

Tabla 3- 28 Conversión del direccionamiento Modbus para los códigos de función FC 01, 05 y 15

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC
De *0* a *2047*	A partir de la marca M *1000*.0
De *2048* a *2559*	A partir de la salida A *256*.0

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#5	Código de función
+2.0	Dirección de bit	WORD	W#16#0019	Dirección de bit
+4.0	Estado de bit	WORD	W#16#FF00	Estado de bit

El esclavo tiene que devolver el telegrama de solicitud al maestro sin modificarlo (eco).

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#5	Código de función
+2.0	Dirección de bit	WORD	W#16#0019	Dirección de bit
+4.0	Estado de bit	WORD	W#16#FF00	Estado de bit

Cálculo de direcciones:

La dirección Modbus "coil_address" 0809 Hex (2057 decimal) se encuentra en el área de salidas:

Byte de salida	=((coil_address- *ccccc*) / 8)	+ *ooooo*
	=((2057 - *2048*) / 8)	+ *256*
	=257	

El número de bit restante da:

Nº bit restante	=((coil_address- *ccccc*)) % 8)	[Módulo 8]
	=((2057 -*2048*) % 8)	
	= 1 ;	

Se accede a la salida A 257.1.

Ejemplos adicionales

Pueden consultarse más ejemplos de acceso a marcas y salidas en el FC 01.

3.6.11 Código de función 06 – Preset Single Register

Finalidad y estructura

El código de función 06 - Preset Single Register presenta las siguientes características:

Función	Esta función permite escribir una palabra de datos en un bloque de datos de la CPU con el sistema maestro Modbus.				
Telegrama de solicitud	ADDR	FUNC	start_register	DATA value (High, Low)	CRC
Telegrama de respuesta	ADDR	FUNC	start_register	DATA value (High, Low)	CRC
LEN en bytes	6				

start_register

La dirección de registro Modbus "start_register" es interpretada por el driver del modo siguiente:

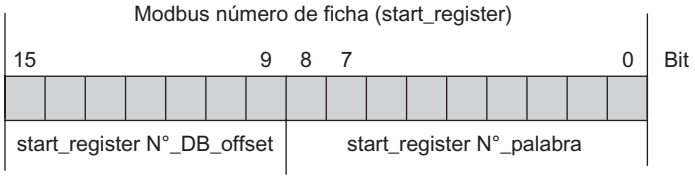


Figura 3-14 Interpretación del número de registro Modbus

Para seguir creando direcciones, el FB81(S_MODB) utiliza el número de DB básico (desde DB xxxxx) registrado en el DB de conversión para los FC 03, 06 y 16.

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza en dos pasos:

Acceso a SIMATIC	Fórmula de conversión
DB de bloque de datos (DB resultante)	=(número de DB básico xxxxx+start_register Nº_DB_offset)
Palabra de datos DBW	=(start_register-Nº_palabra*2)

Si se conoce el DB resultante que debe leerse, la dirección de Modbus start_register que se requiere en el sistema maestro puede calcularse según la fórmula siguiente:

start_register = ((DB resultante - número de DB básico) * 512) + (palabra de datos_DBW / 2)

Para ello sólo deben tomarse como base números de datos pares.

DATA Value

DATA Value (valor de registro) puede ser cualquier valor.

Ejemplo de aplicación para la parametrización:

Tabla 3- 29 Conversión del direccionamiento Modbus para los códigos de función FC 03, 06 y 16

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC
0	A partir del bloque de datos DB *800* (número de DB básico)

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#6	Código de función
+2.0	Dirección de registro	WORD	W#16#0180	Dirección de registro
+4.0	Valor de registro	WORD	W#16#3E7F	Valor de registro

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#6	Código de función
+2.0	Dirección de registro	WORD	W#16#0180	Dirección de registro
+4.0	Valor de registro	WORD	W#16#3E7F	Valor de registro

Cálculo de direcciones:

La dirección Modbus "start_register" 0180 Hex (384 decimal) se interpreta del modo siguiente:

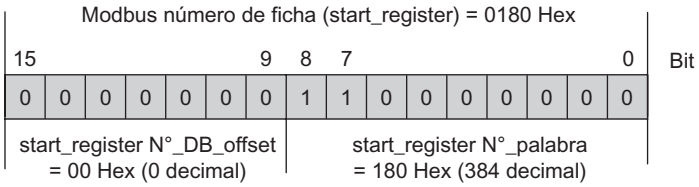


Figura 3-15 Interpretación del número de registro Modbus 0180 Hex

DB de bloque de datos (DB resultante) =(Número de DB básico *xxxxx*+start_register Nº_DB_Offset)
= (*800* + 0)
= 800 ;

Palabra de datos DBW =(start_register Nº_palabra *2)
= (384 * 2)
= 768 ;

Se accede al DB 800, palabra de datos DBW 768.

Ejemplos adicionales

Encontrará más ejemplos de acceso en el FC 03.

3.6.12 Código de función 08 – Loop Back Diagnostic Test

Finalidad y estructura

El código de función 08 – Loop Back Diagnostic Test presenta las siguientes características:

Función	Esta función sirve para comprobar el enlace de comunicación. No afecta a la CPU S7 ni tampoco a los programas y datos de usuario. El driver devuelve el telegrama recibido al sistema maestro de forma totalmente independiente.				
Telegrama de solicitud	ADDR	FUNC	Código de diagnóstico (High, Low)	Datos de test	CRC
Telegrama de respuesta	ADDR	FUNC	Código de diagnóstico (High, Low)	Datos de test	CRC
Código de diagnóstico	Sólo se soporta el código de diagnóstico 0000.				
Datos de test	Cada valor (16 bits).				
LEN en bytes	6				

Ejemplo de aplicación

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#8	Código de función
+2.0	Código de diagnóstico	WORD	B#16#0000	Código de diagnóstico
+4.0	Valor de registro	WORD	B#16#A5C3	Valor de test

DB de destino RCV

La tabla muestra el contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#8	Código de función
+2.0	Código de diagnóstico	WORD	B#16#0000	Código de diagnóstico
+4.0	Valor de test	WORD	B#16#A5C3	Valor de test

3.6.13 Código de función 15 – Force Multiple Coils

Finalidad y estructura

El código de función 15 - Force Multiple Coil presenta las siguientes características:

Función	Esta función permite escribir varios bits en las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro Modbus.							
Telegrama de solicitud	ADDR	FUNC	start_adress		quantity	byte_count N	n-DATA	CRC
Telegrama de respuesta	ADDR	FUNC	start_adress			n Byte DATA		CRC
LEN en bytes	> 6							

start_address

La dirección de bit Modbus "start_address" es interpretada por el driver del modo siguiente:

FB81(S_MODB) comprueba si "start_address" se encuentra en una de las áreas que se indicaron en el DB de conversión para los FC 01, 05, 15 (de/a: marcas, salidas, temporizadores, contadores).

Si la dirección de bits de Modbusstart_address se encuentra en el área	**se accede al área de memoria SIMATIC**	
De *aaaaa* a *bbbbb*	A partir de la marca	M *uuuu*.0
De *ccccc* a *ddddd*	A partir de la salida	A *ooooo*.0

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza del modo siguiente:

Acceso empezando con SIMATIC	**Fórmula de conversión**	
Byte de marca	=((start_address- *cccc*) / 8)	+ *uuuu*
Byte de salida	=((start_address- *aaaa*) / 8)	+*oooo*

Acceso a marcas y salidas

Al acceder a las áreas SIMATIC de marcas y salidas, se calcula el número de bit restante y se utiliza para direccionar el bit en cuestión dentro del byte de marca o de salida.

Acceso a temporizadores y contadores.

Con este código de función FC 15 no se permite acceder a las áreas SIMATIC de temporizadores y contadores y el driver rechaza los intentos de acceso con un telegrama de error.

Quantity

Como Quantity (número de bits) se admite cualquier valor entre 1 y 1696.

DATA

El campo DATA contiene estados de bits (cualquier valor).

Ejemplo de aplicación

Tabla 3- 30 Conversión del direccionamiento Modbus para los códigos de función FC 01, 05 y 15

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC
De *0* a *2047*	A partir de la marca M *1000*.0
De *2048* a *2559*	A partir de la salida A *256*.0

Acción

El sistema maestro Modbus quiere escribir en las marcas M 1144.0 ... M 1144.7 y M 1145.0 ... M 1145.3 los siguientes estados de bit:

Marcas	7	6	5	4	3	2	1	0	Bit
M 1144	ON	ON	OFF	OFF	ON	ON	OFF	ON	

Marcas	7	6	5	4	3	2	1	0	Bit
M 1145	-	-	-	-	ON	OFF	OFF	ON	

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#0F	Código de función
+2.0	Dirección inicial de bit	WORD	W#16#0058	Dirección inicial de bit
+4.0	Número de bits	INT	10	Número de bits
+6.0	coil_state[1]	WORD	W#16#EFCD	Status Coil 5FH..58H/57H..50H

Cálculo de direcciones:

La dirección Modbus "coil_address" 0480 Hex (1152 decimal) se encuentra en el área de marcas:

Byte de marca =((start_address- *aaaaa*) / 8) + *uuuuu*
=((1152 - *0*) / 8) + *1000*
=1144;

El número de bit restante da:

Nº bit restante =((start_address- *aaaaa*)) % 8) [Módulo 8]
=((1152 -*0*) % 8)
= 0;

Se accede a las marcas empezando por M 1144.0.

Ejemplos adicionales

Pueden consultarse más ejemplos de acceso a marcas y salidas en el FC 01.

3.6.14 Código de función 16 – Preset Multiple Registers

Finalidad y estructura

El código de función 16 - Preset Multiple Registers presenta las siguientes características:

Función	Este código de función permite escribir varias palabras de datos en un bloque de datos de la CPU SIMATIC con el sistema maestro Modbus.						
Telegrama de solicitud	ADDR	FUNC	start_register	quantity	byte_count N	n-DATA (High, Low)	CRC
Telegrama de respuesta	ADDR	FUNC	start_register		quantity		CRC
LEN en bytes	> 6						

start_register

La dirección de registro Modbus "start_register" es interpretada por el driver del modo siguiente:

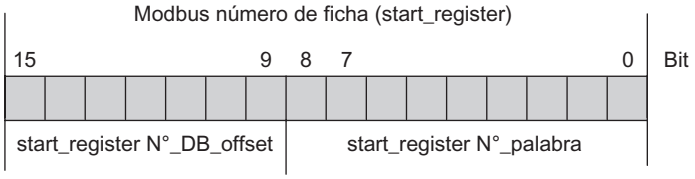


Figura 3-16 Interpretación del número de registro Modbus

Para seguir creando direcciones, el FB81(S_MODB) utiliza el número de DB básico (a partir de DB xxxxx) registrado en el DB de conversión para los FC 03, 06 y 16 durante la parametrización.

A continuación, el cálculo de direcciones para el acceso (conversión de direcciones) se realiza en dos pasos:

Acceso a SIMATIC	Fórmula de conversión
DB de bloque de datos (DB resultante)	=(Número de DB básico *xxxxx*+start_register-Nº_DB_Offset)
Palabra de datos DBW	=(start_register Nº_palabra *2)

Si se conoce el DB resultante que debe escribirse, la dirección Modbus start_register que se requiere en el sistema maestro puede calcularse según la fórmula siguiente:

start_register = ((DB resultante - número de DB básico) * 512) + (palabra de datos_DBW / 2)

Para ello sólo deben tomarse como base números pares de palabras de datos.

Quantity

Como Quantity (número de registros) se admite cualquier valor entre 1 y 109.

DATA (High, Low)

DATA (High, Low) (valor de registro) puede ser cualquier valor. El sistema maestro Modbus quiere escribir en las palabras de datos DBW 100, DBW 102, DBW 104 de DB 800 los valores CD09 Hex, DE1A Hex y EF2B Hex.

Ejemplo de aplicación

Tabla 3- 31 Conversión del direccionamiento Modbus para los códigos de función FC 03, 06 y 16

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC	
0	A partir del bloque de datos (número de DB básico)	DB *800*

DB de origen SEND

La tabla muestra la estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#10	Código de función
+2.0	Dirección inicial de registro	WORD	W#16#0060	Dirección inicial de registro
+4.0	Número de registros	INT	3	Número de registros
+6.0	reg_data[1]	WORD	W#16#41A1	Datos de registro
+8.0	reg_data[2]	WORD	W#16#42A2	Datos de registro
+10.0	reg_data[3]	WORD	W#16#43A3	Datos de registro

Cálculo de direcciones:

La dirección Modbus "start_register" 0032 Hex (50 decimal) se interpreta del modo siguiente:

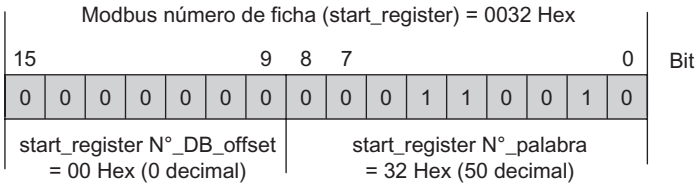


Figura 3-17 Interpretación del número de registro Modbus 0032 Hex

DB de bloque de datos (DB resultante)	=(Número de DB básico *xxxxx*+start_register Nº_DB_Offset) =(*800* + 0) = 800 ;
Palabra de datos DBW	=(start_register Nº_palabra *2) =(50 * 2) = 100;

Se accede al DB 800, palabra de datos DBW 100.

Ejemplos adicionales

Encontrará más ejemplos de acceso en el FC 03.

3.6.15 Conversión del código de función orientada al bit

Código de función 02

El código de función 02 orientado a bits permite acceder en modo de sólo lectura a las áreas de memoria SIMATIC de marcas y entradas.

Mediante el DB de conversión puede determinarse desde y hasta qué dirección Modbus se accederá a las marcas y entradas. Asimismo, puede parametrizarse a partir de qué elemento de datos del área de memoria SIMATIC debe empezar el acceso.

Las áreas de direccionamiento Modbus y las áreas de memoria SIMATIC del FC 02 pueden seleccionarse independientemente de los FC 01, 05 y 15.

Tabla 3- 32 Áreas de direcciones

Dirección Modbus en el telegrama de transferencia	Área de memoria SIMATIC	
De kkkkk	Marcas	De
A lllll		M vvvv.0
De nnnnn	Entradas	De
A rrrr		E sssss.0

3.6.16 Conversión de códigos de función orientada en fichas

Códigos de función 03, 06,16

Los códigos de función orientados a registros 03, 06 y 16 permiten acceder al área de memoria SIMATIC de bloques de datos tanto en modo de lectura como de escritura.

El cálculo del número de bloque de datos deseado se realiza en dos pasos.

1. Con la interfaz de parametrización puede determinarse un número de DB básico. Este DB básico es el primer DB al que puede accederse.
2. La dirección Modbus transferida en el telegrama Start_Register (número de registro) se interpreta del modo siguiente:

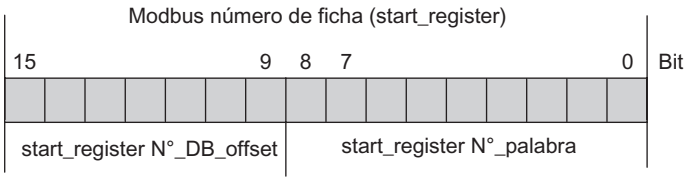


Figura 3-18 Interpretación del número de registro Modbus

Número de DB resultante

El número de DB resultante, al que se puede acceder resulta de:
Número de DB básico + número de DB offset.

De este modo puede accederse a un área de 128 bloques de datos conexos dentro de toda el área de bloques de datos direccionables (65535 DBs).

Número de palabra en el DB

Dentro de cada bloque de datos puede direccionarse el área de DBW 0 a DBW 1022 a través del número de palabra.

El driver interpretar por palabras los DBs organizados por bytes a partir de la estructura básica.

Particularidades del código de función 04

El código de función 04 orientado a registros únicamente permite un acceso de sólo lectura al área de memoria SIMATIC de bloques de datos.

El modo en que se produce el acceso es el mismo que el descrito para los códigos de función 03, 06 y 16.

Para el código de función 04 puede parametrizarse libremente un número de DB básico propio con el DB de conversión. De este modo puede seleccionarse una segunda área independiente que abarque 128 DBs.

No obstante, sólo puede accederse a estos DBs en modo de sólo lectura.

3.6.17 Habilitar/bloquear accesos de escritura

Códigos de función 05, 06, 15, 16

Para los códigos de función de escritura 05, 06, 15 y 16 puede bloquearse o limitarse el acceso a las correspondientes áreas de memoria SIMATIC.

Con el DB de conversión puede definirse un área que esté habilitada para accesos de escritura por parte del sistema maestro Modbus.

Si el maestro intenta acceder a áreas de memoria SIMATIC que se encuentran fuera del área habilitada, el acceso se rechaza con un telegrama de error (excepción). La tabla muestra la habilitación de accesos de escritura.

Tabla 3- 33 Habilitar/bloquear accesos de escritura

38.0	DB_Number _FC_04	WORD	W#16#0	W#16#2	DB	04
40.0	DB_Min	WORD	W#16#0	W#16#1	Número de DB más pequeño empleado	Límites
42.0	DB_Max	WORD	W#16#0	W#16#6	Número de DB más grande empleado	
44.0	M_Min	WORD	W#16#0	W#16#1F4	Marca más pequeña empleada	
46.0	M_Max	WORD	W#16#0	W#16#4B0	Marca más grande empleada	
48.0	Q_Min	WORD	W#16#0	W#16#0	Salida más pequeña empleada	
50.0	Q_Max	WORD	W#16#0	W#16#64	Salida más grande empleada	

3.6.18 Conversión de las direcciones de Modbus para funciones de bit

Códigos de función 01, 05, 15

Los códigos de función orientados a bits 01, 05 y 15 permiten acceder a las áreas de memoria SIMATIC de marcas, salidas, temporizadores y contadores tanto en modo de lectura como de escritura.

Los temporizadores y contadores están protegidos contra escritura con FC01.

Mediante el DB de conversión puede determinarse desde y hasta qué dirección Modbus se accederá a las salidas, los temporizadores y los contadores. Asimismo, puede parametrizarse a partir de qué elemento de datos del área de memoria SIMATIC debe empezar el acceso.

Sinopsis de 01, 05, 15

Tabla 3- 34 Conversión del direccionamiento Modbus para los códigos de función FC 01, 05 y 15

DB de parámetros		Entrada	Significado
Área SIMATIC de marcas			
Dirección Modbus en el telegrama de transferencia (número de bit)	De aaaa	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
	A bbbb	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC Marcas (Marca)	A partir de M uuuuu.0	De 0 a 65535 (decimal)	A partir de este byte de marca
Área SIMATIC de salidas			
Dirección Modbus en el telegrama de transferencia (número de bit)	De cccc	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
	A dddd	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC *Salidas* (Número de byte de salida)	A partir de A ooooo.0	De 0 a 65535 (decimal)	A partir de este byte de salida

DB de parámetros		Entrada	Significado
Área SIMATIC de temporizadores			
Dirección Modbus en el telegrama de transferencia (número de bit)	De eeee	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
	A ffff	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC de *temporizadores* (Número del temporizador)	A partir de A ttttt	De 0 a 65535 (decimal)	A partir de este temporizador (= palabra de 16 bits)
Área SIMATIC de contadores			
Dirección Modbus en el telegrama de transferencia (número de bit)	De gggg	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
	A hhhh	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC de *contadores* (Número del contador)	A partir de Z zzzzz	De 0 a 65535 (decimal)	A partir de este contador (= palabra de 16 bits)

Dirección Modbus "De/A"

Con la dirección "de" puede parametrizarse la dirección Modbus con la que debe empezar el área correspondiente, p. ej. marcas, salidas, etc. (= primer número de bit del área).

Con la dirección "A" puede parametrizarse la dirección Modbus con la que debe finalizar el área correspondiente, p. ej. marcas, salidas, etc. (= último número de bit del área).

Las direcciones "De"/"A" hacen referencia a la dirección Modbus en el telegrama de transferencia (números de bit a partir de 0) con los códigos de función FC 01, 05 y 15.

Las diferentes áreas "De/A" no pueden solaparse.

Puede haber huecos entre las diferentes áreas "De/A".

Área de memoria SIMATIC "A partir de"

Con el dato "A partir de" puede determinarse el principio del área SIMATIC en la que se representa el área Modbus "De/A" (= primer número de byte de marca, de byte de salida, de temporizador y de contador del área SIMATIC).

Ejemplo de FC 01, 05, 15

Tabla 3- 35 Conversión del direccionamiento Modbus para los códigos de función FC 01, 05 y 15

DB de parámetros		Entrada	Significado
Área SIMATIC de marcas			
Dirección Modbus en el telegrama de transferencia	De 0	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
(número de bit)	A 2047	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC Marcas (Marca)	A partir de M 1000.0	De 0 a 65535 (decimal)	A partir de este byte de marca
Área SIMATIC de salidas			
Dirección Modbus en el telegrama de transferencia	De 2048	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
(número de bit)	A 2559	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC *Salidas* (Número de byte de salida)	A partir de A 256.0	De 0 a 65535 (decimal)	A partir de este byte de salida
Área SIMATIC de temporizadores			
Dirección Modbus en el telegrama de transferencia	De 4096	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
(número de bit)	A 4255	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC de *temporizadores* (Número del temporizador)	A partir de T 100	De 0 a 65535 (decimal)	A partir de este temporizador (= palabra de 16 bits)
Área SIMATIC de contadores			
Dirección Modbus en el telegrama de transferencia	De 4256	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
(número de bit)	A 4415	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC de *contadores* (Número del contador)	A partir de Z 120	De 0 a 65535 (decimal)	A partir de este contador (= palabra de 16 bits)

Las direcciones Modbus de 0 a 2047 acceden a las marcas SIMATIC a partir de la marca M 1000.0. Es decir, Por ejemplo, longitud del área = 2048 bits = 256 bytes, esto significa que el último bit de marca = M 1255.7.

Las direcciones Modbus de 2048 a 2559 acceden a las salidas de SIMATIC a partir de la salida A 256.0. Es decir, Por ejemplo, longitud del área = 512 bits = 64 bytes, esto significa que el último bit de salida = A 319.7.

Las direcciones Modbus de 4096 a 4255 acceden a los temporizadores de SIMATIC a partir del temporizador T 100. Es decir, longitud del área = 160 bits = 10 palabras, esto significa que el último temporizador = T 109.

Las direcciones Modbus de 4256 a 4415 acceden a los contadores SIMATIC a partir del contador Z 120. Es decir, longitud del área = 160 bits = 10 palabras, esto significa que el último contadores = Z 129.

Sinopsis de FC02

Tabla 3- 36 Conversión del direccionamiento Modbus para FC 02

DB de parámetros		Entrada	Significado
Área SIMATIC de marcas			
Dirección Modbus en el telegrama de transferencia (número de bit)	De	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
	A	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área SIMATIC de *marcas*	A partir de	De 0 a 65535 (decimal)	A partir de este byte de marca
Área SIMATIC de entradas			
Dirección Modbus en el telegrama de transferencia (número de bit)	De	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
	A	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC de *entradas* (Número de byte de entrada)	A partir de E	De 0 a 65535 (decimal)	A partir de este byte de entrada

Dirección Modbus "De/A"

Con la dirección "De" puede parametrizarse la dirección Modbus con la que debe empezar el área correspondiente, p. ej. marcas, entradas, etc. (= primer número de bit del área).

Con la dirección "A" puede parametrizarse la dirección Modbus con la que debe terminar el área correspondiente (= último número de bit del área).

Las direcciones "De/A" hacen referencia a la dirección Modbus en el telegrama de transferencia (números de bit a partir de 0) con el código de función FC 02.

Las diferentes áreas "De/A" no pueden solaparse.

Puede haber huecos entre las diferentes áreas "De/A".

Área de memoria SIMATIC "A partir de"

Con el dato "A partir de" puede determinarse el principio del área SIMATIC en la que se representa el área Modbus "De/A" (= primer número de byte de marca y de entrada del área SIMATIC).

Ejemplo de FC 02

Tabla 3- 37 Conversión del direccionamiento Modbus para FC 02

DB de parámetros		Entrada	Significado
Área SIMATIC de marcas			
Dirección Modbus en el telegrama de transferencia	De *0*	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
(número de bit)	A *4095*	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área SIMATIC de *marcas*	A partir de M *0*.0	De 0 a 65535 (decimal)	A partir de este byte de marca
Área SIMATIC de entradas			
Dirección Modbus en el telegrama de transferencia	De *4096*	De 0 a 65535 (decimal)	Empezando por esta dirección Modbus
(número de bit)	A *5119*	De 0 a 65535 (decimal)	Inclusive esta dirección Modbus
Área de memoria SIMATIC de *entradas* (Número de byte de entrada)	A partir de E *128*.0	De 0 a 65535 (decimal)	A partir de este byte de entrada

Las direcciones Modbus de 0 a 4095 acceden a las marcas SIMATIC a partir de la marca M 0.0: Es decir, longitud del área = 4096 bits = 512 bytes, esto significa que el último bit de marca = M 511.7.

Las direcciones Modbus de 4096 a 5119 acceden a las entradas SIMATIC a partir de la entrada E 128.0: Es decir, longitud del área = 1024 bits = 128 bytes, esto significa que el último bit de entrada = E 255.7.

Nota

La entrada del valor "a partir de la marca" es totalmente independiente de la entrada "a partir de la marca" de los códigos de función 01, 05 y 15.

Así pues, con el FC 02 puede emplearse una segunda área de marcas SIMATIC (sólo lectura) totalmente independiente de la primera.

3.6.19 Conversión de las direcciones de MODBUS para funciones de ficha

Sinopsis de FC 03, 06, 16

Tabla 3- 38 Conversión del direccionamiento Modbus para FC 03, 06, 16

DB de parámetros		Entrada	Significado
Área SIMATIC de bloques de datos			
Dirección Modbus = 0 en el telegrama de transferencia (Número de registro) significa acceso a:			
Área de memoria SIMATIC de *bloque de datos*	A partir de DB	De 1 a 65535 (decimal)	A partir de este bloque de datos A partir de DBW 0 (= número de DB básico)

A partir de DB

Con la entrada "A partir de DB" puede determinarse el primer bloque de datos del área SIMATIC a la que se accederá (= número de DB básico).

Se accede a este DB cuando el número de registro del telegrama Modbus tiene el valor 0, empezando por la palabra de datos DBW 0.

Los números de registro Modbus superiores acceden a las siguientes palabras de datos/a los siguientes bloques de datos.

Pueden direccionarse hasta 127 DBs consecutivos.

Para acceder a los diferentes DBs consecutivos, el driver interpreta los bits 9 - 15 del número de registro Modbus.

Ejemplo de aplicación

Tabla 3- 39 Conversión del direccionamiento Modbus para FC 03, 06, 16

DB de parámetros		Entrada	Significado
Área SIMATIC de bloques de datos			
Dirección Modbus = 0 en el telegrama de transferencia (Número de registro) significa acceso a:			
Área de memoria SIMATIC de *bloque de datos*	A partir de DB 800	De 1 a 65535 (decimal)	A partir de este bloque de datos A partir de DBW 0 (como número de DB básico)

Con la dirección de registro Modbus 0 se accede al bloque de datos 800 a partir del DBW 0 en el sistema SIMATIC.

Las direcciones de registro Modbus superiores (≥ 512, etc.) acceden a los DBs siguientes, como DB 801, etc.

Sinopsis de FC 04

Tabla 3- 40 Conversión del direccionamiento Modbus para FC 04

DB de parámetros		Entrada	Significado
Área SIMATIC de bloques de datos			
La dirección Modbus = 0 en el telegrama de transferencia (número de registro) significa acceso a:			
Área de memoria SIMATIC de *bloques de datos*	A partir de DB	De 1 a 65535 (decimal)	A partir de este bloque de datos A partir de DBW 0 (como número de DB básico)

A partir de DB

Con la entrada "A partir de DB" puede determinarse el primer bloque de datos del área SIMATIC a la que se accederá (= número de DB básico).

Se accede a este DB cuando el número de registro del telegrama Modbus tiene el valor 0, empezando por la palabra de datos DBW 0.

Los números de registro Modbus superiores acceden a las siguientes palabras de datos/a los siguientes bloques de datos.

Pueden direccionarse hasta 127 DBs consecutivos. Para acceder a los diferentes DBs consecutivos, el driver interpreta los bits 9 - 15 del número de registro Modbus.

Nota

La entrada del valor "A partir de DB" es completamente independiente de la entrada "A partir de DB" de los códigos de función 03, 06 y 16.
Con el FC 04 también es posible utilizar una segunda área de bloques de datos SIMATIC (sólo lectura), completamente independiente de la primera.

Ejemplo de FC 04

Tabla 3- 41 Conversión del direccionamiento Modbus para FC 04

DB de parámetros		Entrada	Significado
Área SIMATIC de bloques de datos			
La dirección Modbus = 0 en el telegrama de transferencia (número de registro) significa acceso a:			
Área de memoria SIMATIC de *bloques de datos*	A partir de DB 1200	De 1 a 65535 (decimal)	A partir de este bloque de datos A partir de DBW 0 (como número de DB básico)

Con la dirección de registro Modbus 0 se accede al bloque de datos 1200 a partir del DBW 0 en el sistema SIMATIC.

Las direcciones de registro Modbus superiores (≥ 512, 1024 etc.) acceden a los DBs siguientes, como DB 1201, 1202, etc.

3.6.20 Límites para funciones de escritura

Sinopsis de FC 05, 06, 16

Tabla 3- 42 Límites SIMATIC para accesos de escritura (FC 05, 06, 16)

DB de parámetros		Entrada	Significado
Bloques de datos DB: Número de DB resultante	DB MIN	De 1 a 65535	Primer DB habilitado
	DB MAX	De 1 a 65535	Ultimo DB habilitado MAX=0 todos los DBs bloqueados
Marca M (Número de byte de marca)	M MIN	De 0 a 65535	Primer byte de marca habilitado
	M MAX	De 1 a 65535	Último byte de marca habilitado MAX=0 todas las marcas bloqueadas
Salidas A (Número de byte de salida)	Q MIN	De 0 a 65535	Primer byte de salida habilitado
	Q MAX	De 1 a 65535	Último byte de salida habilitado MAX=0 todas las salidas bloqueadas

Área de memoria SIMATIC MIN/MAX

En los códigos de función de escritura pueden determinarse límites superiores e inferiores para el acceso (MIN / MAX). El acceso de escritura sólo es posible dentro del área habilitada.

Si se indica 0 como límite superior, toda el área está bloqueada.

Al realizar la selección, tenga en cuenta el tamaño de área correspondiente en SIMATIC, que depende de la CPU.

Si el maestro intenta acceder en modo de escritura a un área que está fuera de los límites superior e inferior, el módulo rechaza el acceso con un telegrama de error.

Los valores MIN/MAX para el área de bloques de datos deben indicarse como números de DB resultantes.

Ejemplo de aplicación de FC 05, 06, 16

Tabla 3- 43 Límites SIMATIC para accesos de escritura (FC 05, 06, 16)

DB de parámetros		Entrada	Significado
Bloques de datos DB: Número de DB resultante	MIN 600	1 a 65535	Primer DB habilitado
	MAX 699	1 a 65535	Último DB habilitado MAX=0 todos los DBs bloqueados
Marca M (Número de byte de marca)	MIN 1000	0 a 65535	Primer byte de marca habilitado
	MAX 1127	1 a 65535	Último byte de marca habilitado MAX=0 todas las marcas bloqueadas
Salidas A (Número de byte de salida)	MIN 256	0 a 65535	Primer byte de salida habilitado
	MAX 319	1 a 65535	Último byte de salida habilitado MAX=0 todas las salidas bloqueadas

Puede accederse a los bloques de datos de SIMATIC DB 600 a DB 699 con códigos de función de escritura (FC 06, 16).

Puede accederse a los bytes de marca SIMATIC MB 1000 a MB 1127 (FC 05, 15) con códigos de función de escritura.

Puede accederse a los bytes de salida SIMATIC AB 256 a AB 319 (FC 05, 15) con códigos de función de escritura.

3.7 Diagnóstico

3.7.1 Posibilidades de diagnóstico

Principio

Con las funciones de diagnóstico del módulo interfaz serie ET 200S Modbus/USS se puede determinar la causa de todos los errores que surjan durante el funcionamiento. Existen las siguientes posibilidades de diagnóstico:

- Diagnóstico con los LEDs de estado del panel frontal del módulo interfaz serie ET 200S Modbus/USS
- Diagnóstico con la salida STATUS de los bloques de función
- Diagnóstico con el diagnóstico de esclavos PROFIBUS

3.7.2 Información de diagnóstico de los LEDs de estado

Función de los LEDs de estado

Los siguientes LEDs de estado se encuentran en el panel frontal del módulo interfaz serie ET 200S Modbus/USS:

- TX (verde): Se enciende cuando el módulo interfaz serie ET 200S Modbus/USS envía datos a través de la interfaz.
- RX (verde): Se enciende cuando el módulo interfaz serie ET 200S Modbus/USS recibe datos a través de la interfaz.
- SF (rojo): El LED de error colectivo indica un posible error:
  - Error de hardware
  - Error de parametrización
  - Rotura de hilo o cable separado entre el módulo interfaz serie ET 200S Modbus/USS y el interlocutor: Se detecta únicamente en conexiones de interfaz RS-422 de diagnóstico de esclavo con el parámetro Preasignación de la línea de recepción = R(A) 5V / R(B) 0V.
  - Error de comunicación (paridad, error de trama, desbordamiento del búfer)

3.7.3 Avisos de diagnóstico de los bloques de función

Estructura de los mensajes de diagnóstico de los bloques de función

Para el diagnóstico de errores, cada bloque de función dispone de un parámetro STATUS. Cada número de mensaje STATUS tiene el mismo significado, independientemente del bloque de función utilizado. La figura siguiente muestra la estructura del parámetro STATUS.

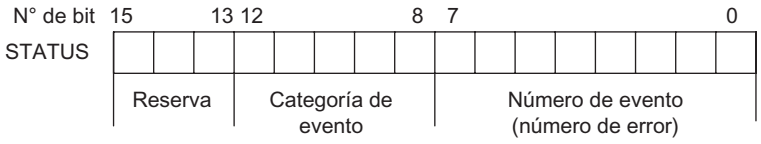


Figura 3-19 Estructura del parámetro STATUS

La figura siguiente muestra a modo de ejemplo el contenido del parámetro STATUS para el evento "Interrupción de petición por rearranque completo, rearranque o reset" (clase de evento $1E_H$, número de evento $0D_H$).

Evento: “Cancelación de petición a causa de rearranque completo, rearranque o reinicialización”

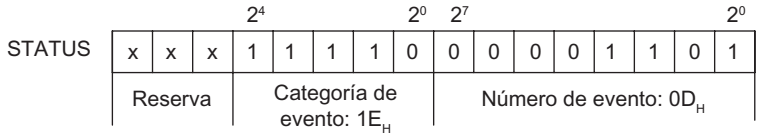


Figura 3-20 Ejemplo: Parámetro STATUS para la clase de evento 1EH, evento 0DH

Llamada de la variable SFCERR

La variable SFCERR contiene más información sobre los errores 14 ($1E\ 0E_H$) y 15 ($1E\ 0F_H$) en la clase de evento 30.

Cargue la variable SFCERR desde el DB de instancia del bloque de función correspondiente.

Los mensajes de error registrados en la variable SFCERR se describen en el apartado dedicado a las funciones de sistema SFC14 "DPRD_DAT" y SFC15 "DPWR_DAT" del manual de referencia *Software de sistema para S7-300 y S7-400. Funciones estándar y funciones de sistema*.

Significado de los mensajes de diagnóstico de los bloques de función

Las tablas siguientes describen las clases de evento, las definiciones de los números de evento y la solución recomendada para cada condición de error.

Tabla 3- 44 Clase de evento 2 (0x02 Hex): Error al procesar una petición de la CPU

Clase de evento 2 (0x02 Hex): "Error al inicializar"			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
(02) 01_H	1	Sin parametrización (válida).	Asigne parámetros correctos al módulo. Dado el caso, compruebe que el equipo se haya instalado de forma adecuada.

Tabla 3- 45 Clase de evento 5 (05 Hex): Error al procesar una petición de la CPU

Clase de evento 5 (05 Hex): Error al procesar una petición de la CPU			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
(05) 02_H	2	Petición no permitida en este estado operativo del módulo interfaz serie ET 200S Modbus/USS (ejemplo: la interfaz del equipo no está parametrizada).	Evalúe la alarma de diagnóstico y corrija el error correspondiente.
(05) $0E_H$	14	Longitud de telegrama no válida	El telegrama de envío tiene más de 224 bytes. La petición de envío ha sido cancelada por el módulo ET 200S Modbus/USS. Seleccione una longitud de telegrama menor.
(05) 30_H	48	Petición de envío del maestro Modbus rechazada, ya que todavía no se ha consultado la respuesta del interlocutor a la petición de envío de escritura anterior del maestro Modbus.	Después de una petición de envío de lectura del maestro Modbus completada con éxito, hay que recoger del módulo la respuesta del interlocutor antes de iniciar una nueva petición de envío del maestro Modbus.
(05) 51_H	81	Error de proceso de trama en la comunicación entre el módulo interfaz serie ET 200S Modbus/USS y la CPU. El error ha surgido al transferir un telegrama recibido del módulo interfaz serie ET 200S 1SI en la CPU.	El módulo y la CPU han cancelado la transferencia. Repita la petición de recepción. El módulo interfaz serie ET 200S Modbus/USS vuelve a enviar el mensaje recibido.

Tabla 3- 46 Clase de evento 8 (08 Hex): Error de recepción

Clase de evento 8 (08 Hex): Error de recepción			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
(08) 06_H	6	Tiempo de retardo entre caracteres excedido. Dos caracteres consecutivos no se han recibido dentro del tiempo de retardo entre caracteres.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
08 $0A_h$	10	Desbordamiento por exceso del búfer de recepción en el maestro durante la recepción del telegrama de respuesta.	Compruebe las configuraciones de protocolo del esclavo.
(08) $0C_H$	12	Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de rebosamiento).	Los fallos en la línea de transmisión provocan repeticiones del telegrama reduciendo el paso de datos útiles. El riesgo de que se produzca un error no detectado aumenta. Modifique la configuración de su sistema o la disposición de la línea. Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos.
(08) $0D_H$	13	BREAK: La línea de recepción que llega al interlocutor está interrumpida.	Restablezca la conexión o conecte el interlocutor.
(08) 10_H	16	Error de paridad: Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión (rotura de línea) de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.
(08) 11_H	17	Error de trama de caracteres: Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión (rotura de línea) de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.
(08) 12_H	18	Se han recibido más caracteres después de que la interfaz serie haya puesto CTS en OFF.	Parametrice de nuevo el interlocutor o descargue la interfaz de serie más rápidamente.

Clase de evento 8 (08 Hex): Error de recepción			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
08 30_H	48	**Maestro**: Se ha enviado un telegrama de solicitud y ya se ha agotado el tiempo de supervisión de respuesta sin que se haya reconocido el principio de un telegrama de respuesta. **Esclavo**: Broadcast no está permitido con este código de función.	Compruebe si la línea de transferencia está interrumpida (puede ser necesario un análisis de la interfaz). Compruebe que el módulo y el interlocutor tengan la misma configuración para los parámetros de protocolo velocidad de transferencia, número de bits de datos, paridad y número de bits de parada. Compruebe si el valor del tiempo de supervisión de respuesta en PtP_PARAM es suficientemente elevado. Compruebe si la dirección de esclavo indicada existe. El sistema maestro Modbus sólo puede emplear broadcast en los códigos de función habilitados para ello.
08 31_H	49	**Maestro**: El primer carácter del telegrama de respuesta del esclavo es diferente de la dirección de esclavo enviada en el telegrama de solicitud (para el modo normal). **Esclavo**: El código de función recibido no está permitido.	Ha respondido el esclavo erróneo. Compruebe si la línea de transferencia está interrumpida (puede ser necesario un análisis de la interfaz). Este código de función no puede emplearse para este driver.
08 32_H	50	Se ha superado el número máximo de bits o de registros o el número de bits no es divisible por 16 cuando se accede a las áreas de memoria SIMATIC de temporizadores o contadores.	Limite la cantidad máxima de bits a 2040 y la cantidad máxima de registros a 127. Acceso a temporizadores/contadores SIMATIC sólo en intervalos de 16 bits.
08 33_H	51	El número de bits o de registros en el código de función FC 15/16 y el elemento de telegrama byte_count no concuerdan.	Corrija el número de bits/registros o byte_count.
08 34_H	52	Se ha detectado una codificación de bits no permitida para "activar bit"/"desactivar bit".	Utilice para FC05 sólo las codificaciones 0000Hex o FF00Hex.
08 35_H	53	Se ha detectado un subcódigo de diagnóstico no permitido (diferente a 0000Hex) en el código de función FC 08 "Loop Back Test".	Utilice para FC08 sólo el subcódigo 0000Hex.

Clase de evento 8 (08 Hex): Error de recepción			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
08 36_H	54	El valor creado internamente para la suma de verificación CRC16 no coincide con la suma de verificación CRC recibida.	Compruebe la formación de la suma de verificación CRC en el sistema maestro Modbus.
08 37_H	55	Error en proceso de telegrama: el sistema maestro Modbus ha enviado un nuevo telegrama de solicitud antes de que el driver haya transferido el último telegrama de respuesta.	Aumente el timeout del telegrama de respuesta del esclavo en el sistema maestro Modbus.
08 50_H	80	La longitud del telegrama de recepción es superior a 224 bytes o mayor que la longitud de telegrama parametrizada.	Adaptar la longitud de telegrama del interlocutor.

Tabla 3- 47 Clase de evento 14 (0E Hex) Errores generales de procesamiento <Parametrización>

Clase de evento 14 (0E Hex) Errores generales de procesamiento <Parametrización>			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
0E 20_H	32	El número de bits de datos de este acoplamiento debe ser 8. El driver no está listo.	Corrija la parametrización del driver.
0E 21_H	33	El factor de multiplicación parametrizado para el tiempo de retardo entre caracteres no está entre 1 y 10. El driver trabaja con la configuración estándar 1.	Corrija la parametrización del driver.
0E 22_H	34	El modo de operación parametrizado del driver no es válido. Debe indicarse "Modo normal" o "Modo con supresión de frecuencias perturbadoras". El driver no está listo.	Corrija la parametrización del driver.

Clase de evento 14 (0E Hex) Errores generales de procesamiento <Parametrización>			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
0E 23_H	35	**Maestro**: Se ha ajustado un valor no válido para el tiempo de supervisión de respuesta: Los valores válidos están entre 50 y 655000-ms. El driver no está listo. **Esclavo**: Se ha ajustado un valor no válido para la dirección de esclavo: La dirección de esclavo 0 no es válida. El driver no está listo.	Corrija la parametrización del driver. Corrija la parametrización del driver.
0E $2E_H$	46	Se ha producido un error al leer el archivo de parámetros de la interfaz. El driver no está listo.	Reinicie el maestro (Mains_ON).

Tabla 3- 48 Clase de evento 14 (0E Hex) Errores generales de procesamiento <Procesamiento de una petición S_SEND>

Clase de evento 14 (0E Hex) Errores generales de procesamiento <Procesamiento de una petición S_SEND>			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
0E 40_H	64	El valor indicado para LEN en S_SEND es demasiado pequeño.	La longitud mínima es de 2 bytes.
0E 41_H	65	El valor indicado para LEN en S_SEND es demasiado pequeño. Es necesaria una longitud mayor para el código de función transferido.	La longitud mínima para este código de función es de 6 bytes.
0E 42_H	66	El código de función transferido no es válido.	Utilice sólo códigos de función permitidos.
0E 43_H	67	Dirección del esclavo 0 (= broadcast) no permitida en este código de función.	Utilice la dirección de esclavo 0 sólo en los códigos de función adecuados.
0E 44_H	68	El valor del "número de bits" transferido no está entre 1 y 2040.	El "número de bits" debe estar entre 1 y 2040.
0E 45_H	69	El valor del "número de registros" transferido no está entre 1 y 127.	El "número de registros" debe estar entre 1 y 127.
0E46_H	70	Código de función 15 ó 16: Los valores del "número de bits" o "número de registros" transferidos no están entre 1 y 2040 o entre 1 y 127, respectivamente.	El "número de bits" o "número de registros" debe estar entre 1 y 2040 o entre 1 y 127, respectivamente.

Clase de evento 14 (0E Hex) Errores generales de procesamiento <Procesamiento de una petición S_SEND>			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
0E 47_H	71	Código de función 15 ó 16: LEN para S_SEND no coincide con el "número de bits" o "número de registros" transmitido. LEN es demasiado pequeño.	Aumente LEN para SEND hasta que se transmita una cantidad suficiente de datos de usuario al módulo. Debe transmitirse una cantidad mayor de datos de usuario al módulo debido al "número de bits" o "número de registros".
0E 48_H	72	Código de función 5: El código indicado en el DB de origen SEND para "activar bit" (FF00H) o para "desactivar bit" (0000H) es erróneo.	Los únicos códigos válidos son "activar bit" (FF00H), "desactivar bit" o 0000H.
0E 49_H	73	Código de función 8: El código indicado en el DB de origen SEND para "código de diagnóstico" es erróneo.	El único código válido es "código de diagnóstico" 0000H.
0E $4A_H$	74	La longitud de este código de función es superior a la longitud máxima.	En el manual encontrará los datos de longitud máxima para cada código de función.

Tabla 3- 49 Clase de evento 14 (0E Hex) Errores generales de procesamiento <Valoración de recepción>

Clase de evento 14 (0E Hex) Errores generales de procesamiento <Valoración de recepción>			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
0E 50_H	80	El maestro ha recibido una respuesta sin enviar.	Hay un esclavo u otro maestro en la red. Compruebe si la línea de transferencia está interrumpida (puede ser necesario un análisis de la interfaz).
0E 51_H	81	Código de función erróneo: El código de función recibido en el telegrama de respuesta es distinto del código de función enviado.	Compruebe el equipo esclavo.
0E 52_H	82	Rebase por defecto de bytes: El número de caracteres recibidos es inferior al que resultaría del contador de bytes del telegrama de respuesta o bien es inferior al esperado para este código de función.	Compruebe el equipo esclavo.
0E 53_H	83	Rebase por exceso de bytes: El número de caracteres recibidos es superior al que resultaría del contador de bytes del telegrama de respuesta o bien es superior al esperado para este código de función.	Compruebe el equipo esclavo.
0E 54_H	84	Error de contador de bytes: El contador de bytes recibido en el telegrama de respuesta es demasiado pequeño.	Compruebe el equipo esclavo.
0E 55_H	85	Error de contador de bytes: El contador de bytes recibido en el telegrama de respuesta es incorrecto.	Compruebe el equipo esclavo.
0E 56_H	86	Error de eco: Los datos del telegrama de respuesta (número de bits, ...) repetidos por el esclavo son distintos de los datos enviados en el telegrama de solicitud.	Compruebe el equipo esclavo.
0E 57_H	87	Error en la comprobación CRC: Al comprobar la suma de verificación CRC16 del telegrama de respuesta del esclavo se ha producido un error.	Compruebe el equipo esclavo.

Tabla 3- 50 Clase de evento 14 (0E Hex) Errores generales de procesamiento <Recepción mensaje de código de excepción>

Clase de evento 14 (0E Hex) Errores generales de procesamiento <Recepción mensaje de código de excepción>			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
0E 61_H	97	Telegrama de respuesta con código de excepción 01: función no válida	Véase el manual del equipo esclavo.
0E 62_H	98	Telegrama de respuesta con código de excepción 02: dirección de datos no válida	Véase el manual del equipo esclavo.
0E 63_H	99	Telegrama de respuesta con código de excepción 03: valor de datos no válido	Véase el manual del equipo esclavo.
0E 64_H	100	Telegrama de respuesta con código de excepción 04: fallo en el equipo correspondiente	Véase el manual del equipo esclavo.
0E 65_H	101	Telegrama de respuesta con código de excepción 05: acuse	Véase el manual del equipo esclavo.
0E 66_H	102	Telegrama de respuesta con código de excepción 06: ocupado, telegrama rechazado	Véase el manual del equipo esclavo.
0E 67_H	103	Telegrama de respuesta con código de excepción 07: acuse negativo	Véase el manual del equipo esclavo.

Tabla 3- 51 Clase de evento 30 (1E Hex): Error durante la comunicación entre la interfaz serie y la CPU

Clase de evento 30 (1E Hex): Error durante la comunicación entre la interfaz serie y la CPU			
Número de evento	**Número de evento (decimal)**	**Evento**	**Solución**
(1E) $0D_H$	13	"Interrupción del trabajo por rearranque completo, rearranque o reset"	
(1E) $0E_H$	14	Error estático al llamar la SFC DP_RDDAT. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
(1E) $0F_H$	15	Error estático al llamar la SFC DP_WRDAT. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
(1E) 10_H	16	Error estático al llamar la SFC RD_LGADR. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
(1E) 11H	17	Error estático al llamar la SFC RDSYSST. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
(1E) 20_H	32	Parámetro fuera del rango.	Introduzca un parámetro dentro del rango válido para el bloque de función.
(1E) 41_H	65	El número de bytes indicado en el parámetro LEN de los FB no está permitido	Respete el rango de 1 a 256 bytes.

Evaluación de la variable SFCERR

Obtendrá más información sobre los errores ocurridos (1E) $0E_H$, (1E) $0F_H$, (1E) 10_H y (1E) 11_H de la clase de evento 30 a través de la variable SFCERR.

La variable SFCERR se puede cargar desde el DB de instancia del correspondiente bloque de función.

Los mensajes de error registrados en la variable SFCERR se encuentran en las funciones de sistema "DPRD_DAT", SFC15 "DPWR_DAT" y RD_LGADR en el manual de referencia *Software de sistema para S7-300 y S7-400. Funciones estándar y funciones de sistema*.

3.7.4 Diagnóstico de esclavos PROFIBUS

Introducción

El diagnóstico de esclavos se atiene a la norma EN 50170, volumen 2, PROFIBUS. Dependiendo del maestro DP, el diagnóstico puede leerse con STEP 5 o STEP 7 para todos los esclavos DP que cumplan con la norma mencionada.

El diagnóstico de esclavos PROFIBUS abarca el diagnóstico y estado de módulos y el diagnóstico de canal. Encontrará información detallada sobre el diagnóstico de esclavos DP en el manual *Sistema de periferia descentralizada ET 200S, 6ES7 151-1AA10-8AA0*).

Diagnóstico de canal

El diagnóstico de canal proporciona información sobre errores de canal de módulos y comienza tras el estado del módulo. La tabla siguiente lista los tipos de error de canal.

Tabla 3- 52 Tipos de error de canal del módulo interfaz serie ET 200S Modbus/USS

Evento (tipo de error)	Descripción	Medida recomendada
00110: Rotura de hilo	Cable roto o suelto.	Compruebe el cableado de los bornes. Compruebe el cable del interlocutor.
00111: Rebase por exceso	Desbordamiento del búfer; desbordamiento de la longitud del mensaje	El FB S_RCV debe ser llamado con mayor frecuencia.
01000: Rebase por defecto	Se ha enviado un mensaje con longitud 0.	Compruebe el motivo por el que el interlocutor envía telegramas sin datos útiles.
01001: Fallo	Ha aparecido un error interno del módulo.	Cambie el módulo.
10000: Error de parametrización	Módulo no parametrizado.	Corrija la parametrización.
10110: Error de mensaje	Error de trama, error de paridad	Compruebe los ajustes de comunicación.

3.7.5 Funciones de diagnóstico del esclavo Modbus

ERROR_NR y ERROR_INFO

El FB de comunicación Modbus tiene los dos parámetros de salida siguientes, con los que se visualizan los errores surgidos:

- Parámetro ERROR_NR
- Parámetro ERROR_INFO

Los errores surgidos se visualizan en la salida ERROR_NR. En la salida ERROR_INFO se emite información más detallada sobre el error de ERROR_NR.

Borrado de errores

Los errores se borran en un flanco positivo de START. Asimismo, el usuario puede borrar las indicaciones de error en cualquier momento, cuando sea necesario.

Códigos de error de FB

Los códigos de error 1 - 99 tienen el significado siguiente:

- ERROR_No 1 - 9

  Error al inicializar FB y CP

  Los números de error 1...9 indican que la inicialización ha finalizado con errores. El parámetro START_ERROR es 1.

  No es posible una comunicación de MODBUS con el sistema maestro.

- ERROR_No 10 - 19

  Error al procesar un código de función

  Los números de error 10...19 indican que se ha producido un error al procesar un código de función. El módulo ha enviado una petición de procesamiento no válida al FB de comunicación.

  El error también se comunica al driver.

  Las peticiones de procesamiento posteriores se seguirán procesando.

- ERROR_No 90 - 99

  Otros errores

  Se ha producido un error de procesamiento.

  El error no se comunica al driver.

  Las peticiones de procesamiento posteriores se seguirán procesando.

3.7.6 Errores

Lista de los números de error

Tabla 3- 53 Errores durante la inicialización

Número de error (decimal)	ERROR_INFO	Evento	Solución
0	0	Ningún error.	
1	SFC51->RET_VAL	Error al leer la lista de estado del sistema con la SFC51.	Analizar RET_VAL en ERROR_INFO, eliminar la causa.
2	S_SEND->STATUS, S_RCV->STATUS	Timeout en la inicialización del módulo o error al inicializar el módulo (error en la petición S_SEND).	Compruebe si para esta interfaz se ha configurado "MODBUS Slave" como protocolo. Compruebe si el "ID" indicado en el FB de comunicación es correcto. Analice ERROR_INFO.

Tabla 3- 54 Error al procesar un código de función

Número de error (decimal)	ERROR_INFO	Evento	Solución
11	Dirección inicial	Dirección inicial no válida transferida por el driver al FB de comunicación.	Compruebe la dirección Modbus del sistema maestro Modbus.
12	Número de registros	Número de registros no válido transferido por el driver al FB de comunicación. Número de registros = 0.	Compruebe el número de registros del sistema maestro Modbus, en caso necesario, reinicie el módulo (Mains_ON).
13	Número de registros	Número de registros no válido transferido por el driver al FB de comunicación: Número de registros > 128.	Compruebe el número de registros del sistema maestro Modbus, en caso necesario, reinicie el módulo (Mains_ON).
14	Marcas M – Dirección final	Intento de acceso al área de memoria SIMATIC de las marcas más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.

Número de error (decimal)	ERROR_INFO	Evento	Solución
15	Salidas A – Dirección final Entradas E – Dirección final	Intento de acceso al área de memoria SIMATIC de las salidas más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.
16	Temporizadores T – Dirección final	Intento de acceso al área de memoria SIMATIC de los temporizadores más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.
17	Contadores Z – Dirección final	Intento de acceso al área de memoria SIMATIC de los contadores más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.
18	0	Área de memoria SIMATIC no válida transferida por el driver al FB de comunicación.	En caso necesario, reinicie el módulo (Mains_ON).
19		Error al acceder a la periferia SIMATIC.	Compruebe si la periferia necesaria existe y no tiene errores.
20	DB#	El DB no existe.	Incluya el DB en su proyecto.
21	DB#	Longitud de DB no válida	Aumente la longitud de DB.
22	DB#	DB# por debajo del valor mínimo de DB.	Cambie el valor mínimo de DB.
23	DB#	DB# por encima del valor máximo de DB.	Cambie el valor máximo de DB.
24	Dirección de marca	Marca por debajo del límite mínimo.	Cambie los límites mínimos de las marcas en el DB de conversión.
25	Dirección de marca	Marca por encima del límite máximo.	Cambie los límites máximos de las marcas en el DB de conversión.

Número de error (decimal)	ERROR_INFO	Evento	Solución
26	Dirección de salida	Salida por debajo del límite mínimo.	Cambie los límites mínimos de las salidas en el DB de conversión.
27	Dirección de salida	Salida por encima del límite máximo.	Cambie los límites máximos de las salidas en el DB de conversión.

Tabla 3- 55 Otros errores

Número de error (decimal)	ERROR_INFO	Evento	Solución
90	S_SEND-> STATUS	Error al transferir un telegrama de acuse al driver con S_SEND.	Analice la información de STATUS.
94	S_RCV->STATUS	Error al leer de SYSTAT con S_RCV (STATUS).	Analice la información de STATUS.

3.8 Maestro de USS

3.8.1 ¿Qué es el maestro de USS?

Introducción

Con el protocolo USS, el usuario puede configurar la comunicación de bus serie entre el módulo ET 200S Modbus/USS como maestro y varios sistemas esclavos. Los accionamientos de Siemens pueden emplearse como esclavos en el bus USS.

Características del protocolo USS

El protocolo USS presenta las siguientes características principales:

- Compatibilidad con el acoplamiento RS485 con capacidad multipunto
- Modelo de acceso maestro-esclavo
- Sistema con un maestro
- 32 estaciones como máximo (máx. 31 esclavos)
- Funcionamiento con variables o longitudes de telegrama predeterminadas
- Telegramas sencillos y fiables
- El mismo funcionamiento de bus que para PROFIBUS (DIN 19245 1ª parte)
- Interfaz de datos con el convertidor de accionamiento básico conforme a los accionamientos PROFIL con diferentes velocidades. Es decir, que al utilizar USS, la información se transmite al accionamiento como con PROFIBUS-DP.
- Empleo para arranque, mantenimiento y automatización

3.8.2 Protocolo de USS

Introducción

El protocolo USS es un protocolo sencillo de transferencia de datos serie, que está adaptado a las necesidades de la tecnología de accionamientos.

El protocolo USS define una técnica de acceso de acuerdo con el principio maestroesclavo para la comunicación a través de un bus serie. Es posible conectar al bus un maestro y un máximo de 31 esclavos. El maestro selecciona los diferentes esclavos en el telegrama con la ayuda de un carácter de dirección. Un esclavo no puede enviar nunca sin que el maestro lo haya activado antes. Esto significa que no es posible una transferencia de datos directa entre esclavos. La comunicación funciona en modo semidúplex. La función del maestro no puede transferirse. El sistema USS sólo dispone de un maestro.

Estructura de telegrama

Cada telegrama empieza con un carácter de inicio (STX), seguido de la información sobre la longitud (LGE) y el byte de dirección (ADR). A continuación sigue el campo de datos. El telegrama finaliza con el carácter de comprobación de bloque (BCC).

STX	LGE	ADR	1	2	...	N	BCC

En el caso de datos de un bloque de datos de red compuestos por una palabra (16 bits), primero se envía el byte High y después el byte Low. En el caso de datos de palabras dobles, primero se envía la palabra High y, a continuación, la palabra Low.

El protocolo no identifica ninguna tarea en los campos de datos.

Codificación de datos

Los datos se codifican del modo siguiente:

- STX: 1 byte, inicio de texto, 02H
- LGE: 1 byte, contiene la longitud de telegrama como número binario
- ADR: 1 byte, contiene la dirección del esclavo y el tipo de telegrama en código binario
- Campos de datos: Un byte cada uno, el contenido depende de la tarea
- BCC: 1 byte, carácter de comprobación de bloque

Procedimiento de transmisión de datos

El maestro se ocupa de la transferencia de datos cíclica en telegramas. El maestro direcciona sucesivamente todas las estaciones esclavo con un telegrama de tarea. Las estaciones en cuestión responden con un telegrama de respuesta. De acuerdo con el procedimiento maestroesclavo, el esclavo debe enviar el telegrama de respuesta al maestro después de haber recibido el telegrama de tarea. Sólo entonces el maestro puede comunicarse con el siguiente esclavo.

Estructura general del bloque de datos de red

El bloque de datos de red está dividido en dos secciones: Parámetros (PKW) y datos de proceso (PZD).

STX	LGE	ADR	Parámetros (PKW)	Datos de proceso (PZD)	BCC

- **Área de parámetros** (PKW)

  El área PKW gestiona la transferencia de parámetros entre dos interlocutores (p. ej. autómata y accionamiento). Esto incluye, por ejemplo, leer y escribir valores de parámetros así como leer descripciones de parámetros y el texto correspondiente. La interfaz de PKW suele contener tareas para manejo e indicadores, mantenimiento y diagnóstico.

- **Área de datos de proceso** (PZD)

  El área PZD incluye señales necesarias para la automatización:

  – Palabras de control y valores consigna del maestro al esclavo

  – Palabras de estado y valores reales del esclavo al maestro

  Los accionamientos esclavos definen el contenido del área de parámetros y del área de datos de proceso. Encontrará más información al respecto en la documentación del accionamiento.

3.8.3 Configuración y parametrización

Configuración y parametrización

Tabla 3- 56 Parámetros del maestro USS

Parámetro	Descripción	Rango de valores	Valor por defecto
Alarma de diagnóstico	Indique si el módulo crea una alarma de diagnóstico cuando surge un error grave.	• No • Sí	No
Activar reconocimiento BREAK	Si se produce una ruptura de la línea o no se ha conectado ningún cable de interfaz, el módulo notifica el mensaje de error "Break".	• No • Sí	No
Tipo de interfaz	Indique la interfaz eléctrica que debe emplearse.	• RS-232 • RS-485 (semidúplex)	RS-485 (semidúplex)
Preasignación dúplex y semidúplex de la línea de recepción	Indique la preasignación de la línea de recepción en el modo de operación RS-485. No en el modo de operación RS-232C. La configuración "Nivel invertido" sólo es requerida en caso de repuesto para asegurar la compatibilidad.	R(A) 5V / R(B) 0V R(A) 0V / R(B) 5V Valor invertido Sin	R(A) 0V / R(B) 5V
Velocidad de transmisión	Seleccione la velocidad de la transmisión de datos en bits por segundo.	• 110 • 300 • 600 • 1200 • 2400 • 4800 • 9600 • 19200 • 38400 • 57600 • 76800 • 115200	9600

Nota

Tenga en cuenta también las indicaciones recogidas en los temas Datos de identificación (Página 63) y Cargar actualizaciones de firmware a posteriori (Página 65).

3.8.4 Resumen de funciones

Procedimiento de transmisión de datos de red

Los bloques gestionan cíclicamente la transferencia de datos de red con un máximo de 31 accionamientos esclavos de acuerdo con el orden indicado en la lista de sondeo (DB de parametrización). Cada esclavo sólo tiene una petición activa. El usuario guarda los datos de red para cada esclavo en un bloque de datos (bloque de datos de red) y los llama desde allí. Se transfieren al procesador de comunicaciones a través de otra área de datos (DB de procesador de comunicaciones) de acuerdo con la definición de programa en la lista de sondeo y se llaman desde allí.

Para este procedimiento se requieren dos llamadas de funciones (un bloque de envío y un bloque de recepción). Una tercera función soporta la creación y la asignación de valores predeterminados de los bloques de datos necesarios para la comunicación.

Características y prestaciones:

- Creación de áreas de datos para la comunicación según la configuración de bus
- Valores predeterminados de la lista de sondeo
- Estructura de telegramas de acuerdo con la especificación USS
- El intercambio de datos de red puede parametrizarse de acuerdo con la estructura necesaria de datos de red
- Ejecución y supervisión de peticiones de PKW
- Gestión de informes sobre modificación de parámetros
- Supervisión de todo el sistema y eliminación de fallos

Pueden emplearse diferentes estructuras de datos de red para enviar datos de red.

Según sea la estructura seleccionada, los datos de red disponen de un área PZD para los datos de proceso y de un área PKW para el procesamiento de parámetros.

En el área PKW, el maestro puede leer y escribir los valores de parámetros y el esclavo puede visualizar modificaciones de parámetros mediante los informes correspondientes.

El área PZD contiene las señales necesarias para el control de procesos, p. ej. palabras de control, el valor consigna del maestro al esclavo, así como palabras de estado y valores reales del esclavo al maestro.

El orden correcto de las llamadas de funciones es: S_USST, S_SEND, S_RCV, S_USSR. Esto es importante porque las salidas de las funciones S_SEND y S_RCV sólo son válidas en el ciclo actual del sistema de automatización.

La figura siguiente muestra el intercambio de datos entre el programa de usuario y el esclavo USS.

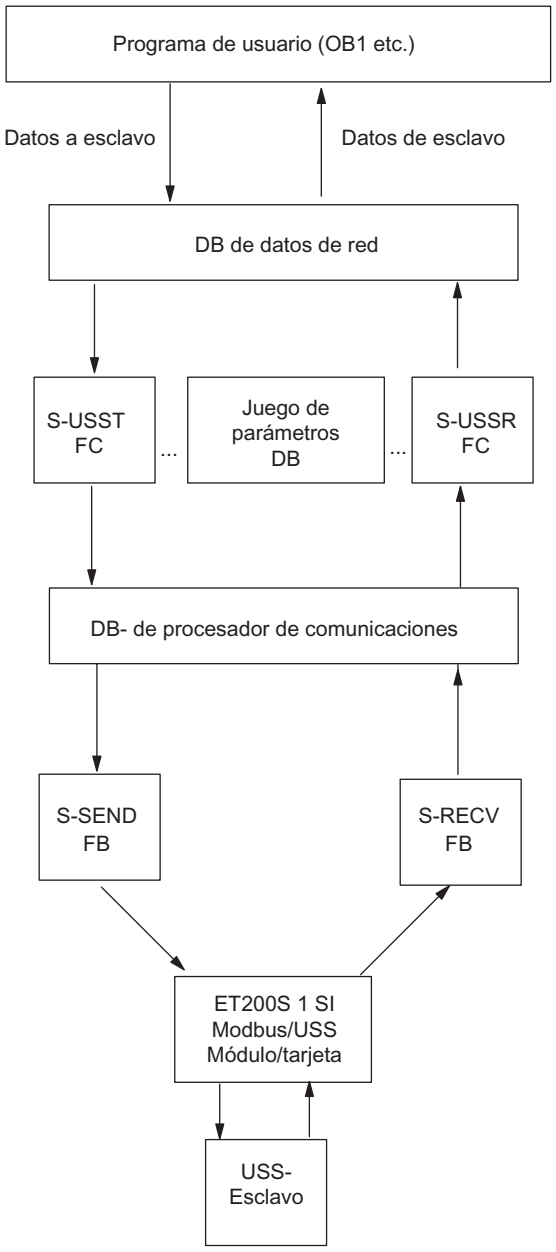


Figura 3-21 Intercambio de datos entre el programa de usuario y el esclavo USS

3.8.5 FC17 S_USST: Enviar datos a un esclavo

Descripción

La FC S_USST ejecuta la transferencia de los datos de red (PZD y, si procede, datos PKW) a los esclavos según la estructura de datos de red utilizados.

La FC toma la parametrización del esclavo actual de la lista de sondeo (DB de parametrización) y envía los datos desde el DB de datos de red. Evalúa la palabra de control de comunicación del esclavo actual (lanzamiento de una petición de PKW o acuse de un informe de modificación de parámetros), completa los datos de envío de USS y los transfiere al búfer de envío del DB de procesador de comunicaciones. Finalmente, lanza la transferencia de los datos de red al esclavo mediante el FB S_SEND.

Si la función detecta un error de parametrización en el DB de parametrización, se guarda una señal de error en el byte PAFE 2 (byte de error de parametrización) del DB de datos de red.

La FC17 se llama una vez en cada ciclo del sistema de automatización.

Estructura del programa de S_USST

La figura siguiente muestra la estructura del programa de S_USST.

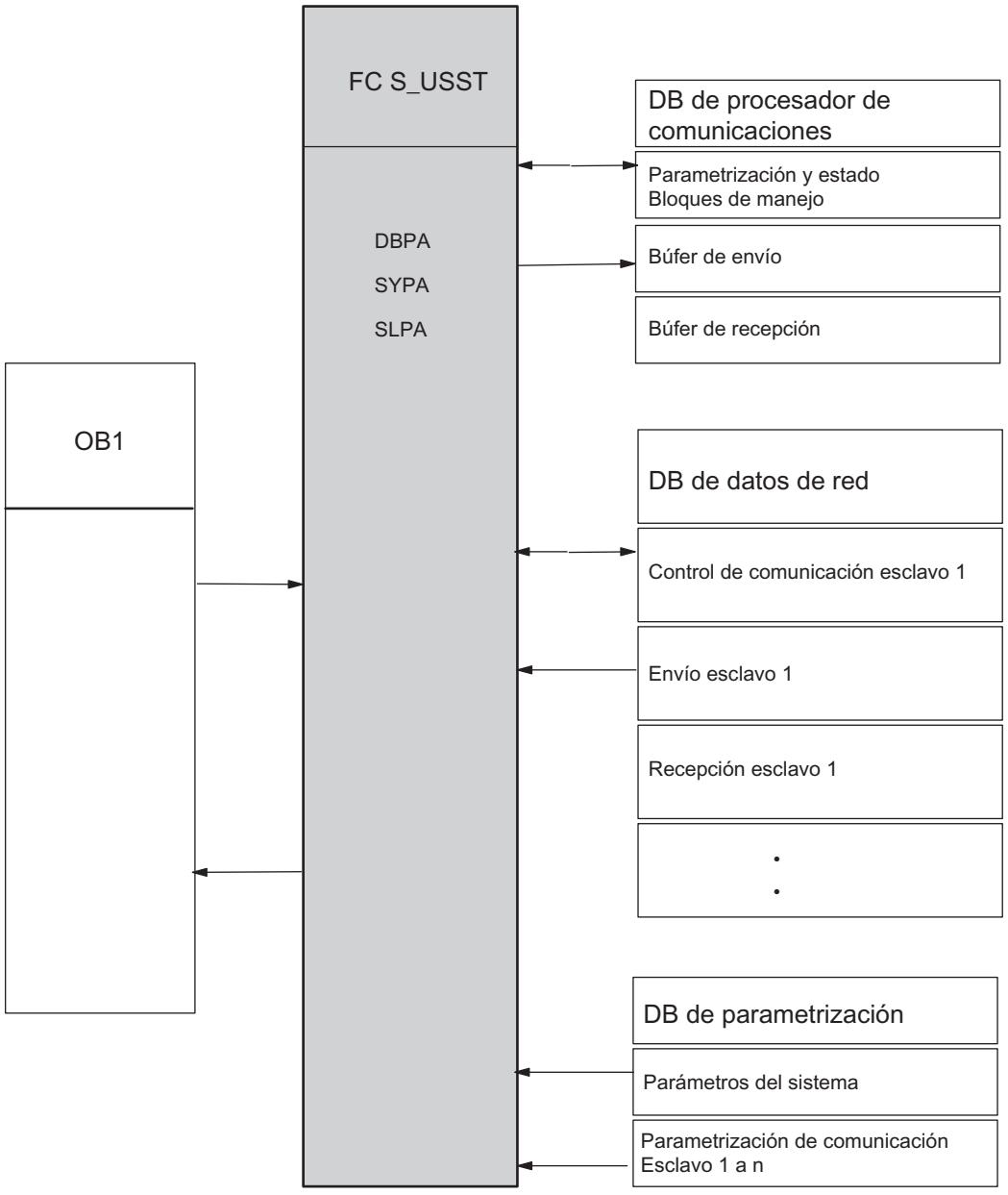


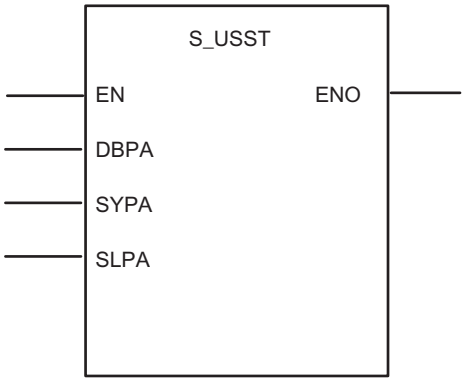
Figura 3-22 Funciones de diagnóstico de esclavo Modbus

Tabla 3- 57 Representación en AWL y KOP

Representación AWL

```
CALL                S_USST
      DBPA =
      SYPA =
      SLPA =
```

Representación KOP



Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Parámetro FC17 S_USST

La tabla lista los parámetros de la FC S_USST.

Tabla 3- 58 Parámetros de la FC S_USST

Nombre	Tipo	Tipo de datos	Descripción	Comentario
DBPA	INPUT	INT	Número de bloque del DB de parametrización	Específico de la CPU (Cero no está permitido)
SYPA	INPUT	INT	Dirección inicial de los parámetros de sistema en el DB de parametrización	0 <= SYPA <= 8174
SLPA	INPUT	INT	Dirección inicial de los parámetros del esclavo en el DB de parametrización	0 <= SLPA <= 8184

3.8.6 FC18 S_USSR: Recibir datos de un esclavo

Descripción

La FC S_USSR gestiona la recepción de datos de red (datos de PZD y, dado el caso, también de PKW) de los esclavos según la estructura de datos de red utilizada.

La FC toma la parametrización del esclavo actual de la lista de sondeo (DB de parametrización) y evalúa la palabra de estado del bloque TRANSMIT.

Si la petición actual ha finalizado sin errores (bit 9 = 0 en la palabra de estado de comunicación del DB de datos de red), los datos entrantes se transfieren del búfer de recepción del DB de procesador de comunicaciones al DB de datos de red y se evalúan. A continuación, se actualiza la palabra de estado de comunicación en el DB de datos de red.

Si la petición actual no ha finalizado sin errores (bit 9 = 1 en la palabra de estado de comunicación del DB de datos de red), el búfer de recepción del DB de procesador de comunicaciones no acepta los datos del esclavo actual. La FC18 indica esta situación en la palabra de estado de comunicación del DB de datos de red y registra la causa del error en la palabra de errores de comunicación.

Si el bloque detecta un error de parametrización en el DB de parametrización, se guarda una señal de error en el byte PAFE 1 del DB de datos de red.

La FC18 se llama una vez en cada ciclo del sistema de automatización.

Estructura del programa de S_USSR

La figura siguiente muestra la estructura del programa de S_USSR.

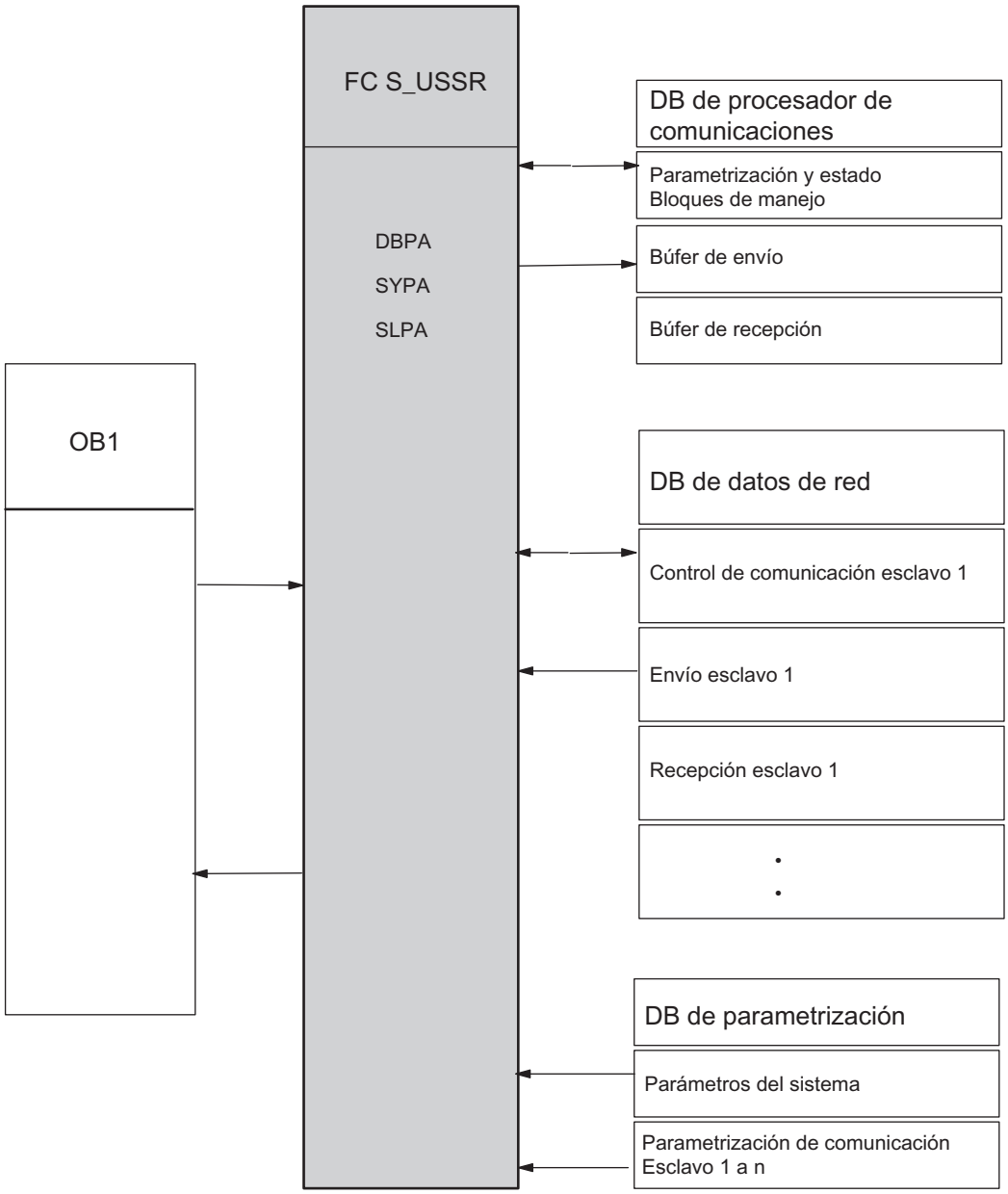
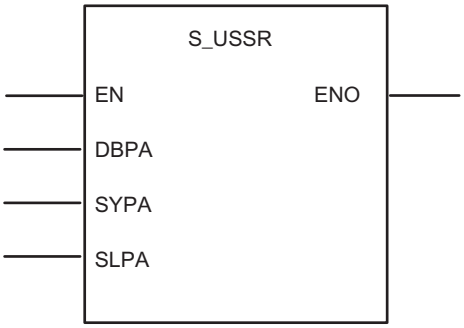


Figura 3-23 Estructura del programa de S_USSR

Tabla 3- 59 Representación en AWL y KOP

Representación AWL	**Representación KOP**
CALL S_USSR DBPA = SYPA = SLPA =	

Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Parámetros de la FC18 S_USSR

La tabla lista los parámetros de la FC S_USSR.

Tabla 3- 60 Parámetros de la FC S_USSR

Nombre	Tipo	Tipo de datos	Descripción	Comentario
DBPA	INPUT	INT	Número de bloque del DB de parametrización	Específico de la CPU (Cero no está permitido)
SYPA	INPUT	INT	Dirección inicial de los parámetros de sistema en el DB de parametrización	0 <= SYPA <= 8174
SLPA	INPUT	INT	Dirección inicial de los parámetros del esclavo en el DB de parametrización	0 <= SLPA <= 8184

Los parámetros de la FC U_USST se corresponden con los parámetros de la FC S_USSR. Ambas funciones acceden a la misma parametrización (parámetros del sistema y del esclavo) en el DB de parametrización y, por ello, deben parametrizarse de forma idéntica.

3.8.7 FC19 S_USSI: Inicialización

Descripción

La FC S_USSI es una función opcional.

Si se llama esta FC al arrancar el sistema S7, se generan los bloques de datos de procesador de comunicaciones, de datos de red y de parametrización necesarios para la comunicación. Asimismo, se asignan valores predeterminados a DBPA. La FC S_USSI sirve para generar el área de datos mencionada y asignarle valores predeterminados, sólo cuando la estructura de los datos de red es igual para todos los esclavos.

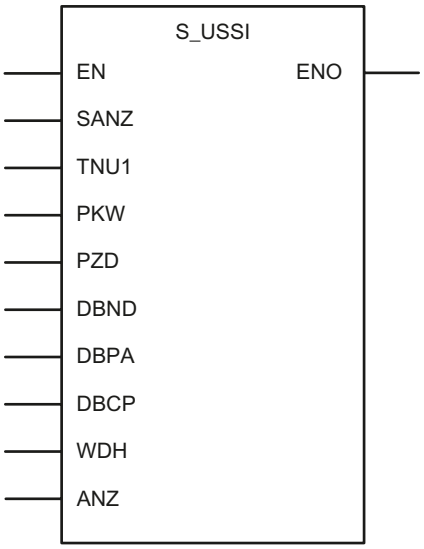
Cuando se llama, la FC comprueba primero la plausibilidad de su parametrización en relación con el número de esclavos, la estructura de los datos de red, el número de estación de inicio y las repeticiones de PKW. Si el bloque detecta un error, no se ejecuta ni la generación ni la asignación de valores predeterminados de los bloques de datos. La CPU pasa al estado operativo STOP y el usuario obtiene un mensaje de error a través del byte de error de la FC S_USSI. Una vez solucionado el error de parametrización, antes de reiniciar deben borrarse todos los bloques de datos generados.

Después de verificar la plausibilidad, el bloque comprueba si los bloques de datos que deben generarse ya existen:

- Si los bloques de datos que deben generarse aún no existen, se crean y se asignan valores predeterminados a DBPA.
- Si ya existen, se comprueba la longitud de cada bloque. Si el DB es lo suficientemente largo, se vuelve a crear la asignación de valores predeterminados del DB de parametrización y se borra el contenido del DB de datos de red y del DB de procesador de comunicaciones. Si un DB es demasiado corto, la CPU pasa al estado operativo STOP. El usuario reconoce el DB defectuoso en el byte de visualización de la FC S_USSI. Para solucionar el error, deben borrarse por completo los tres bloques de datos. Cuando se realiza un rearranque completo, se crean de nuevo y se asignan valores predeterminados al DB de parametrización.

S_USSI tiene que llamarse una vez durante el arranque del sistema (OB100).

Tabla 3- 61 Representación en AWL y KOP

Representación AWL		**Representación KOP**
CALL	S_USSI	
SANZ	:=	
TNU1	:=	
PKW	:=	
PZD	:=	
DBND	:=	
DBPA	:=	
DBCP	:=	
WDH	:=	
ANZ	:=	

Nota

Los parámetros EN y ENO sólo aparecen en la representación gráfica (con KOP o FUP). Para poder procesar estos parámetros, el compilador utiliza el resultado binario RB.

El resultado binario RB se aplica al estado de señal "1" si el bloque se ha finalizado sin errores. Si hay un error, el resultado binario RB pasa a "0".

Parámetros de la FC19 S_USSI

La tabla lista los parámetros de la FC S_USSI.

Tabla 3- 62 Parámetros de la FC S_USSI

Nombre	Tipo	Tipo de datos	Descripción	Comentario
SANZ	INPUT	INT	Número de esclavos con la misma estructura de datos de red (parámetros del sistema en DBPA)	1 <= SANZ <= 31
TNU1	INPUT	INT	Número de estación de inicio (número de equipo)	0 <= TNU1 <= 31
PKW	INPUT	INT	PKW, cantidad	Número de palabras de la interfaz de PKW 0, 3 ó 4
PZD	INPUT	INT	PZD, cantidad	Número de palabras de la interfaz PZD 0 <= PZD <= 16
DBND	INPUT	INT	Número de DB de datos de red	Específico de la CPU (no se admite el cero)
DBPA	INPUT	INT	Número del DB de parametrización	Específico de la CPU (no se admite el cero)
DBCP	INPUT	INT	Número del DB de procesador de comunicaciones	Específico de la CPU (no se admite el cero)
WDH	INPUT	INT	Número de repeticiones permitidas de una petición de PKW	0 <= WDH <= 32767
ANZ	OUTPUT	BYTE	Byte de error	0: Ningún error 1: Número de esclavos excesivo 2: Datos no permitidos para estructura de datos de red 3: DB de parametrización demasiado corto 4: DB de datos de red demasiado corto 5: Error de número de equipo 6: DB de procesador de comunicaciones demasiado corto 7: Libre 8: Contador de repeticiones: Valor incorrecto

3.8.8 DB de datos de red

Descripción

Estos bloques de datos pueden crearse con la FC S_USSI al arrancar la CPU o bien pueden ocuparse con valores predeterminados (sólo DBPA) o introducirse manualmente.

El DB de datos de red constituye la interfaz entre el programa de comunicación y el programa de control. El usuario debe preparar este bloque "vacío" con la longitud suficiente. En el búfer de envío del DB de datos de red, asignado al esclavo por el programa de control, sólo se registran los datos de envío para un esclavo. Los datos de respuesta del esclavo se toman desde el búfer de recepción correspondiente (tras la evaluación del bit 9 en la palabra de control de comunicación). Las palabras de estado permiten controlar la comunicación y la palabra de mando, el lanzamiento selectivo de una petición de parametrización.

La interfaz de comunicación contiene los datos siguientes una vez para cada esclavo:

- Datos de comunicación relacionados con el esclavo (control de comunicación, seguimiento, 6 palabras de datos)
- Búfer para la petición de PKW en curso (sólo si existe un área PKW)
- Búfer de envío para datos de red (máximo 20 palabras de datos)
- Búfer de recepción para datos de red (máximo 20 palabras de datos)

La longitud del búfer de envío y de recepción depende de la estructura de datos de red seleccionada. Si falta la interfaz de PKW, se suprime el búfer para la petición de PKW actual.

La longitud total del DB de datos de red necesario depende de la cantidad de esclavos y de la estructura de datos de red empleada.

Número de palabras de datos por esclavo = 2 x (PKW + PZD) + PKW + 6

Con PKW = 0, 3 ó 4 y 0 <= PZD <= 16

Ejemplo: Un accionamiento con un área PKW de 3 palabras y un área PZD de 2 palabras ocupa 19 palabras de datos en el DB de datos de red.

El DB de datos de red tiene 1550 palabras de datos con 31 esclavos y la longitud máxima de datos de red. DBW0 está reservado.

Asignación de datos del esclavo en el DB de datos de red con 4 palabras en el área PKW y de 0 a 16 palabras en el área PZD

DBWn	Palabra de control de comunicación (KSTW)		Control de la comunicación	
DBWn+2	Interna		Seguimiento de la comunicación Estado de error Contador de intentos de PKW Error de parámetro	
DBWn+4	Palabra de estado de comunicación			
DBWn+6	Palabra de error de comunicación			
DBW n+8	Interna			
DBW n+10	Byte PAFE 1, byte PAFE 2			
DBW n+12	ID de parámetro	PKE	Búfer para petición de PKW actual	
DBW n+14	Índice	IND		
DBW n+16	Valor de parámetro 1	PWE1		
DBW n+18	Valor de parámetro 2	PWE2		
DBW n+20	ID de parámetro	PKE	Área PKW	Búfer de envío
DBW n+22	Índice	IND		
DBW n+24	Valor de parámetro 1	PWE1		
DBW n+26	Valor de parámetro 2	PWE2		
DBW n+28	Palabra de mando (STW)	PZD1	Área PZD (máx. 16 palabras PZD)	
DBW n+30	Valor de consigna principal (HSW)	PZD2		
DBW n+32	Valor consigna/palabra de mando adicional	PZD3		
DBW n+34	Valor consigna/palabra de mando adicional	PZD4		
...	...			
DBW n+58	Valor consigna/palabra de mando adicional	PZD16	Área PKW	Búfer de recepción
DBW n+60	ID de parámetro	PKE		
DBW n+62	Índice	IND		
DBW n+64	Valor de parámetro 1	PWE1		
DBW n+66	Valor de parámetro 2	PWE2		
DBW n+68	Palabra de estado (ZSW)	PZD1	Área PZD	
DBW n+70	Valor real principal (HIW)	PZD2		
DBW n+72	Valor real/palabra de estado adicional	PZD3		

DBW n+74	Valor real/palabra de estado adicional	PZD4	(máx. 16 palabras PZD)	
...	...			
DBW n+98	Valor real/palabra de estado adicional	PZD16		
	•			
(n = 2,4,6...)	•			

Nota

Si hay un área PKW, el búfer para las peticiones de PKW actuales y el área PKW del búfer de envío no existen.

Palabra de control de comunicación KSTW (DBW n)

Los bits de la palabra de mando de comunicación coordinan el programa de usuario y la FC S_USST FC.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Inicio de petición de PKW

  El usuario activa el bit 0 cuando hay una nueva petición de PKW nueva en el búfer de envío y debe procesarse. La FC desactiva el bit cuando se ha aceptado la petición de PKW.

- Bit 1: Aceptación del informe de modificación de parámetros

  El usuario activa el bit 1 cuando se ha aceptado el informe de modificación de parámetros. La FC desactiva el bit para acusar la admisión. Tras este acuse, el esclavo prosigue con el procesamiento interrumpido de la petición en curso o transfiere el siguiente informe de modificación de parámetros.

Palabra de estado de comunicación (DBW n+4)

Los bits de la palabra de estado de comunicación son activados por las FCs S_USST y S_USSR.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Petición de PKW en proceso

  La FC S_USST activa el bit 0 cuando se ha aceptado la petición de PKW y el ID de parámetro (PKE) contiene un ID de petición válido. La FC S_USSR desactiva el bit cuando se ha ejecutado la petición de PKW (con o sin errores) o cuando la interfaz de PKW está defectuosa.

- Bit 1: La petición ha finalizado sin errores

  La FC S_USSR activa el bit 1 cuando se ha ejecutado una petición de PKW sin errores. La respuesta debe tomarse del búfer de recepción. La FC S_USST desactiva el bit cuando se lanza una nueva petición de PKW.

  Nota

  Las peticiones de PKW para el esclavo se procesan en el orden de la lista de sondeo (DBPA). Cada esclavo sólo tiene una petición activa. Si hay más de un esclavo registrado en la lista de sondeo, los datos de respuesta para una nueva petición de PKW sólo están disponibles en un flanco positivo del bit 1 (o del bit 2).

- Bit 2: La petición de PKW ha finalizado con errores

  La FC S_USSR activa el bit 2 en caso de ID de respuesta en PKE. El número de error se encuentra en PWE de la respuesta del esclavo. La FC S_USST desactiva el bit cuando se lanza una nueva petición de PKW.

  Nota

  La última petición de PKW que ha transferido el usuario se guarda en la interfaz de envío tras su procesamiento. La transferencia al esclavo se repite hasta que se introduce una nueva petición. De este modo, es posible que se requieran respuestas adicionales en el programa de usuario si la petición de PKW de estado finaliza con errores (bit 2) y con un fallo de interfaz de PKW (bit 4).

- Bit 3: ID de petición de PKW no válido.

  La FC S_USST activa el bit 3 cuando se determina el ID de petición 15 en PKE o cuando se registra el índice 255 en el ID de petición 4. La FC S_USST desactiva el bit cuando se lanza la siguiente petición de PKW con ID de petición válido en PKE.

- Bit 4: Interfaz de PKW con errores (desbordamiento del contador).

  La FC S_USSR activa el bit 4 cuando el esclavo no responde a una petición de PKW tras repetirla un número de veces parametrizable (parámetro WDH en el DB de parametrización) o cuando el ID de respuesta en PKE es 8. La FC S_USSR restaura el bit cuando se lanza y ejecuta correctamente una nueva petición de PKW.

- Bit 5: Los datos de respuesta contienen un informe de modificación de parámetros.

  La FC S_USSR activa el bit 5 cuando existe un informe de modificación de parámetros del esclavo (ID de respuesta 9 - 12 y bit de conversión 11 invertido). La FC S_USST desactiva el bit cuando el usuario acusa el informe de modificación de parámetros (palabra de mando de comunicación, bit 1).

- Bit 6: Fallo de funcionamiento en el esclavo.

  La FC S_USSR FC activa y desactiva el bit 6. La FC evalúa la palabra de estado (bit 3) del esclavo.

- Bit 7: Hay una advertencia del esclavo

  La FC S_USSR FC activa y desactiva el bit 7. La FC evalúa la palabra de estado (bit 7) del esclavo.

- Bit 8: Se requiere control desde el sistema de automatización.

  La FC S_USSR FC activa y desactiva el bit 8. La FC evalúa la palabra de estado (bit 9) y la palabra de mando (bit 10).

- Bit 9: Error colectivo de comunicación.

  La FC S_USSR FC activa y desactiva el bit 9. La FC evalúa las respuestas de los bloques estándar S_SEND y S_RCV y comprueba el telegrama recibido en relación con ADR, STX, BCC y LGE. La FC comunica además que se ha excedido el tiempo de supervisión de telegrama.

  Nota

  Los datos de recepción del DB de datos de red sólo son válidos si el bit 9 = 0.

Estructura de la palabra de error de comunicación (DBW n+6)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Error de direccionamiento (ADR)
- Bit 3: Inicio de telegrama no detectado (el primer carácter no es STX)
- Bit 4: Error de carácter de comprobación de bloque (BCC)
- Bit 6: Error de longitud de telegrama (LGE)

  La FC S_USSR activa los bits 0, 3, 4 y 6 cuando se detecta un error al verificar el telegrama recibido (ADR, STX, BCC, LGE).

- Bit 7: Tiempo de supervisión de telegrama transcurrido

  La FC S_USSR activa el bit 7 cuando el tiempo entre el envío del telegrama por parte del maestro al esclavo y la llegada de la respuesta del esclavo supera el tiempo permitido calculado por el programa (tiempo de supervisión de telegrama).

Los demás bits no se utilizan.

Byte PAFE 1

Mensaje de error de la FC S_USSR, error de parametrización en el DB de parametrización

- Valor 0: No hay fallo
- Valor 1: Datos erróneos para PKW / PZD

Byte PAFE 2

Mensaje de error de la FC S_USST, error de parametrización en el DB de parametrización

- Valor 0: No hay fallo
- Valor 1: Datos erróneos para PKW / PZD

ID de parámetro PKE en el búfer de envío

El usuario debe introducir el número de parámetros (bits 0 a 10) y el ID de petición (bits 12 a 15). El bit de conmutación para el informe de modificación de parámetros (bit 11) es enmascarado por las funciones S_USSR y S_USST.

3.8.9 DB de parametrización

Descripción

El DB de parametrización contiene los parámetros de programa necesarios para controlar la comunicación. El usuario debe crear este bloque y asignar valores predeterminados a la configuración del sistema de comunicación de forma pertinente (S_USSI o manualmente). Los esclavos del bus se procesan en el orden de la entrada en DBPA (lista de sondeo).

Un esclavo también puede registrarse varias veces en el DB de parametrización, con lo que su prioridad aumenta de forma efectiva.

La longitud del DB de parametrización depende del número n de los esclavos a los que debe accederse en un ciclo de bus.
Número de palabras de datos del DB de parametrización = (n x 4) + 5.

Para cada comunicación con un esclavo se requieren 4 palabras de datos y para los parámetros de sistema se ocupan 4 palabras de datos una sola vez. DBW0 está reservado.

DBW 0	Libre	Parámetros del sistema
DBW 2	DBCP	
DBW 4	SANZ	
DBW 6	SLAV	
DBW 8	WDH	
DBW 10	Número de PKW, número de PZD	Comunicación Registro de parámetros esclavo 1
DBW 12	TUN	
DBW 14	DBND	
DBW 16	KSTW	
DBW 18	Número de PKW, número de PZD	Comunicación Registro de parámetros esclavo 2
DBW 20	TUN	
DBW 22	DBND	
DBW 24	KSTW	
	Número de PKW, número de PZD	Comunicación Registro de parámetros esclavo n
	TUN	
	DBND	
DBW (n x 8 + 8)	KSTW	

Parámetros del sistema

DBCP	Número de bloque del DB de procesador de comunicaciones
SANZ	Número total de registros de parámetros de esclavo en el DB de parametrización Si hay que acceder a determinados esclavos con mayor frecuencia que a otros dentro de un ciclo de bus, registre los parámetros de esclavo varias veces en el DB de parametrización. El parámetro de sistema SANZ debe adaptarse de la forma correspondiente.
SLAV	Número correlativo del esclavo actual. Es requerido por la FC S_USST y la FC S_USSR para determinar el registro de parámetros actual. A esta palabra de datos debe asignársele el valor predeterminado 1 De ello se encarga la FC S_USSI, siempre que se utilice.
WDH	Número de repeticiones permitidas de una petición de PKW (rango de valores: 0 a 32767). Si la petición actual de PKW no finaliza dentro del número de registro, se notifica que la interfaz PKW está defectuosa.

Parametrización de comunicación de esclavo:

Número de PKW, número de PZD	Definición de la estructura de datos de red Byte izquierdo: Número de palabras para el área PKW (0, 3, 4) Byte derecho: Número de palabras para el área PZD (0 a 16) Los datos divergentes se reconocen como errores de parametrización (por las FCs S_USST y S_USSR) y se registran en los bytes PAFE 1 y 2 del DB de datos de red.
TUN	Número de estación, que equivale a la dirección de bus ajustada en el accionamiento (0 a 31).
DBND	Número de bloque del DB de datos de red.
KSTW	Dirección de la palabra de mando de comunicación KSTW para el esclavo en el DB de datos de red.

3.8.10 DB del procesador de comunicaciones

Estructura del DB del procesador de comunicaciones

A través de este bloque de datos tiene lugar el intercambio de datos entre la CPU y el módulo de interfaz serie ET 200S Modbus/USS. El usuario debe poner este bloque a disposición con la longitud suficiente. El DB del procesador de comunicaciones debe tener una longitud mínima de 50 palabras (DBW 0 a 98)

<table>
<tr><td>DBW 0</td><td colspan="2">Estado de la comunicación</td><td>ENVIAR y RECIBIR</td></tr>
<tr><td>DBW 2</td><td>Número máximo de ciclos al esperar a recibir</td><td>Contador de ciclos para formar timeout al esperar a recibir</td><td>FC17

FC17</td></tr>
<tr><td>DBW 4</td><td colspan="2">Pausa de inicio medida</td><td>FC17, OB1</td></tr>
<tr><td>DBW 6</td><td colspan="2">Duración dle último ciclo (OB1_MIN_CYCLE)</td><td>
ENVIAR</td></tr>
<tr><td>DBW 8</td><td colspan="2">Longitud del telegrama de envío (LEN)</td><td></td></tr>
<tr><td>DBB10</td><td colspan="2">Libre</td><td></td></tr>
<tr><td>DBB 11
:
:
DBB 54</td><td colspan="2">Búfer de transmisión</td><td>Enviar telegrama a módulo
(La longitud se rige por la estructura de los datos útiles del esclavo actual)</td></tr>
<tr><td>DBB 55
:
:
DBB 98</td><td colspan="2">Búfer de recepción</td><td>Telegrama recibido por el módulo (La longitud se rige por la estructura de los datos útiles del esclavo actual)</td></tr>
</table>

Estado de comunicación DBW0

El DBW0 contiene los siguientes bits:

- Bit 0: Entrada REQ para S_SEND.

  Este bit se pone a 0 cuando se activa el bit 8.
- Bit 1: Entrada R para S_SEND.

  S_USST pone cíclicamente este bit a cero.
- Bit 2: Salida DONE de S_SEND.
- Bit 3: Salida ERROR de S_SEND.
- Bit 4: Entrada EN_R para S_RCV.

  S_USSR pone cíclicamente este bit a 1.
- Bit 5: Entrada R para S_RCV.

  S_USSR pone cíclicamente este bit a 0.
- Bit 6: Salida NDR de S_RCV.
- Bit 7: Salida ERROR de S_RCV.
- Bit 8: Petición en curso (bit DONE de S_SEND guardado).

  S_USST pone cíclicamente este bit a 0.

Duración dle último ciclo DBW6

S_USST mide con este parámetro el tiempo de respuesta de un esclavo. El programa de usuario debe copiar el tiempo de ciclo del sistema de automatización (OB1_MIN_CYCLE) antes de cada llamada del S_USST en este parámetro.

3.9 Propiedades de arranque y estados operativos del módulo interfaz serie ET 200S Modbus/USS

3.9.1 Carga de los datos de configuración y parametrización

Gestionar los datos

Al cerrar la configuración de hardware, los datos se guardan automáticamente en el proyecto STEP 7.

Cargar la configuración y los parámetros

Los datos de configuración y de parámetros pueden cargarse online a la CPU desde la programadora. Con el comando de menú "Sistema de destino > Cargar" se cargan los datos a la CPU.

Los parámetros del módulo se transfieren automáticamente al módulo durante el arranque de la CPU y en cada transición de STOP a RUN, siempre que el módulo esté accesible a través del bus posterior S7-300.

La interfaz de parametrización de la memoria remanente del módulo guarda el código de driver. Por ello no es posible realizar un cambio de módulo sin programadora.

Más información

En el manual de usuario de STEP 7 se describen detalladamente los siguientes procesos:

- El almacenamiento de la configuración y de parámetros
- La carga de la configuración y de parámetros en la CPU
- La lectura, copia e impresión de la configuración y de parámetros

3.9.2 Estados operativos del módulo interfaz serie ET 200S Modbus/USS

Estados operativos

El módulo interfaz serie ET 200S Modbus/USS permite los siguientes estados operativos:

- STOP:

  Cuando el módulo está en estado STOP no hay ningún driver de protocolo activo, y la CPU acusa negativamente todas las peticiones de envío y recepción. El módulo permanece en el estado operativo STOP hasta que se haya eliminado la causa del estado STOP (ejemplo: rotura de hilo o parámetro no válido).

- Reiniciar parámetros:

  Al reiniciar parámetros del módulo se inicializa el driver de protocolo. El LED SF de error colectivo está encendido durante el proceso de reinicialización.

  No es posible ni enviar ni recibir y los telegramas de envío y recepción guardados en el módulo se pierden a causa del reinicio del driver. La comunicación entre el módulo y la CPU se reinicia (se cancelan los telegramas en curso).

  Al final del proceso de reinicialización de los parámetros, el módulo está en estado operativo RUN y está listo para enviar y recibir.

- RUN:

  El módulo procesa las peticiones de envío de la CPU. La CPU pone a disposición para su lectura los telegramas recibidos por el interlocutor.

3.9.3 Propiedades de arranque del módulo interfaz serie ET 200S Modbus/USS

Fases del arranque

El arranque consta de dos fases:

- **Inicialización**: En cuanto el módulo recibe suministro de tensión, se inicializa la interfaz serie, que espera a datos de parametrización de la CPU.
- **Parametrización**: Durante la parametrización, el módulo interfaz serie ET 200S Modbus/USS recibe los parámetros de módulo, que el usuario ha asignado al slot actual en STEP 7.

3.9.4 Comportamiento del módulo interfaz serie ET 200S Modbus/USS en caso de cambio de estado operativo de la CPU

Comportamiento tras el arranque

Una vez que el módulo interfaz serie ET 200S Modbus/USS ha arrancado, se intercambian todos los datos entre la CPU y el módulo mediante los bloques de función.

- **STOP de la CPU**:

  En el estado operativo STOP de la CPU no es posible la comunicación a través de PROFIBUS. Si hay una transmisión de datos en curso entre el módulo interfaz serie ET 200S Modbus/USS y la CPU, tanto si es una petición de envío como de recepción, ésta se cancela y la conexión se reinicia.

- **Arranque de la CPU**:

  En el arranque, la CPU transmite parámetros al módulo.

  Con la parametrización adecuada se puede borrar automáticamente el búfer de recepción del módulo al arrancar la CPU.

- **RUN de la CPU**:

  Con la CPU en estado RUN son posibles, sin restricciones, los modos de envío y recepción. En las primeras ejecuciones de FB después de reiniciar la CPU se sincronizan el módulo y los FBs correspondientes. Sólo entonces se ejecuta un nuevo S_SEND o S_RCV.

Particularidades al enviar telegramas

Los telegramas sólo pueden enviarse en estado operativo RUN.

Si la CPU pasa a STOP durante la transmisión de datos de la CPU al módulo, S_SEND notifica el error (05) 02_H después del rearranque. Para evitar este comportamiento, el programa de usuario puede llamar S_SEND con la entrada RESET desde el OB de arranque.

Nota

El módulo interfaz serie ET 200S Modbus/USS no envía datos al interlocutor hasta que ha recibido todos los datos del módulo.

Particularidades al recibir telegramas

Con STEP 7 puede parametrizarse "Borrar búfer de recepción del módulo en el arranque = sí/no".

- Si ha parametrizado "sí", el búfer de recepción del módulo interfaz serie ET 200S Modbus/USS se borra automáticamente cuando la CPU pasa de STOP a RUN.
- Si ha parametrizado "No", el telegrama se guarda en el búfer de recepción del módulo interfaz serie ET 200S Modbus/USS.

Si la CPU pasa a STOP durante la transmisión de datos de la CPU al módulo interfaz serie ET 200S Modbus/USS, S_RCV notifica el error (05) 02_H después del rearranque. Para evitar este comportamiento, el programa de usuario puede llamar S_SEND con la entrada RESET desde el OB de arranque. En caso de "Borrar búfer de recepción del módulo interfaz serie ET 200S Modbus/USS en el arranque = no", el módulo transfiere de nuevo el telegrama a la CPU.

3.10 Datos técnicos

Datos técnicos generales

Para el módulo interfaz serie ET 200S 1SI Modbus/USS rigen los datos técnicos generales, tal y como se recogen en el capítulo "Especificaciones técnicas generales" del manual *Sistema de periferia descentralizado ET 200S*. Encontrará dicho manual en:

http://www.siemens.com/simatic-tech-doku-portal

Especificaciones técnicas de los protocolos y de la interfaz

Tabla 3- 63 Datos técnicos de protocolos e interfaces del módulo ET 200S Modbus/USS

Datos técnicos generales	
Elementos de indicación	LED verde, TX (enviar) LED verde, RX (recibir) LED rojo, SF (error colectivo)
Drivers de protocolo suministrados	Driver Modbus Driver USS
Velocidades de transmisión con el protocolo Modbus Velocidades de transmisión con el protocolo USS	110, 300, 600, 1.200, 2.400, 4.800, 9.600, 19.200, 38.400, 57.600, 76.800, 115.200
Trama de caracteres (11 bits)	Cantidad de bits por carácter: 8 Número de bits de arranque/parada: 1 ó 2 Paridad: Sin, par, impar, cualquiera
Memoria necesaria para los bloques estándar (FB)	Enviar y recibir: aprox. 4300 bytes
Datos técnicos de la interfaz RS 232C	
Interfaz	RS232C, 8 bornes
Señales RS 232C	TXD, RXD, RTS, CTS, DTR, DSR, DCD, PE Todas aisladas eléctricamente de la fuente de alimentación interna del módulo ET 200S Modbus/USS.
Recorrido máximo de transferencia	15 m

Datos técnicos generales	
Datos técnicos de la interfaz RS-422/485	
Interfaz	RS-422, 5 bornes RS-485, 3 bornes
Señales RS-422 Señales RS-485	TXD (A)-, RXD (A)-, TXD (B)+, RXD (B)+, PE R/T (A), R/T (B), PE Todas aisladas eléctricamente de la fuente de alimentación interna del módulo ET 200S Modbus/USS.
Recorrido máximo de transferencia	1200 m

Datos técnicos de Modbus/USS

Tabla 3- 64 Datos técnicos generales del módulo ET 200S Modbus/USS

Datos técnicos generales	
Dimensiones y peso	
Dimensiones A x A x P (en mm)	15 × 81 × 52
Peso	aprox. 50 g
Datos específicos del módulo	
RS-232C • Número de entradas • Número de salidas	 4 3
RS-422 • Número de pares de entradas • Número de pares de salidas	 1 1
RS-485 • Número de pares E/S	 1
Longitud de cable • Apantallado (RS232C) • Apantallado (RS-422/485)	 máx. 15 m máx. 1200 m
Grado de protección[1]	IEC 801-5

Datos técnicos generales	
Tensiones, intensidades, potenciales	
Tensión nominal de alimentación de la electrónica (L +)	24V DC
• Protección contra cambio de polaridad	Sí
Aislamiento galvánico	
• Entre los canales y el bus posterior	Sí
• Entre los canales y la tensión de alimentación de la electrónica	Sí
• Entre los canales	No
• Entre los canales y PROFIBUSDP	Sí
Aislamiento ensayado con • Canales respecto al bus posterior y la tensión de carga L+ • Tensión de carga L+ respecto al bus posterior	 500 V DC 500 V AC
Fuente de corriente • Del bus posterior • De la fuente de alimentación L+	 máx. 10 mA máx. 80 mA, típ. 20 mA
Pérdidas del módulo	típ. 1,2 W
Estado, alarmas, diagnóstico	
Indicador de estado	LED verde (TX) LED verde (RX)
Funciones de diagnóstico	
• Indicador de error colectivo • Visualización de información de diagnóstico	LED rojo (SF) Posible

Datos técnicos generales	
Salidas	
Salida, área RS232C	± máx. 10 V
• Con carga capacitiva • Protección contra cortocircuito • Corriente de cortocircuito • Tensión en las salidas o entradas a PE (tierra)	máx. 2500 pF Sí aprox. 60 mA máx. 25 V
Salida, RS-422/485	
Resistencia de carga • Protección contra cortocircuito • Corriente de cortocircuito	mín. 50 kΩ Sí aprox. 60 mA
[1] Equipos de protección externos necesarios en las líneas de entradas de la tensión de usuario: • Blitzductor adaptador estándar del perfil soporte • Blitzductor tipo de módulo de protección KT AD24V	

Tiempos de procesamiento

El tiempo de procesamiento completo maestro-esclavo (con tiempo de actualización de datos) puede determinarse de la siguiente manera:

- Tiempo total de procesamiento (t_8) = tiempo procesamiento petición de maestro (t_1) + tiempo de envío petición de maestro (t_2) + tiempo de procesamiento petición de esclavo (t_3) + 1 ciclo de CPU (tiempo para el procesamiento del código de función) (t_4) + tiempo de procesamiento respuesta de esclavo (t_5) + tiempo de envío respuesta de esclavo (t_6) + tiempo de procesamiento respuesta de maestro (t_7)

Tiempo de procesamiento petición/respuesta

La fórmula para calcular el tiempo de envío y recepción es el mismo para maestro y esclavo. Con una transmisión de datos de 8 bytes, el tiempo de envío y recepción puede determinarse del siguiente modo:

- Si el ciclo de la CPU es >> (ciclos E/S + 10 ms),
  entonces el tiempo de procesamiento = 1 ciclo de la CPU cada 7 bytes,
  en cualquier otro caso tiempo de procesamiento = (2 ciclos de CPU + 3 ciclos E/S + 10-ms) 7 bytes en cada caso

Tiempo de envío/recepción para petición/respuesta

El tiempo para enviar o recibir una petición o bien una respuesta se determina de la siguiente manera:

- Tiempo de envío/recepción = 10 ms + velocidad de transmisión multiplicado por el número de caracteres del mensaje

Tabla 3- 65 Ejemplo de tiempo total de procesamiento:

Leer	Velocidad de transmisión	Ciclo E/S	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
10 palabras	9600 bits/s	2 ms	40 ms	12 ms	40 ms	40 ms	160 ms	29 ms	160 ms	483 ms

Índice alfabético

B

Breves instrucciones de puesta en marcha
    Interfaz serie, 13

C

Campo de datos DATA, 144
    Bytecount, 144
    Coil_Start Address, 144
    Number_of_Coils, 144
    Number_of_Registers, 144
    Register_Start Address, 144
Códigos de función, 144
Códigos de función de esclavo, 144, 187
    Código de función 01 - Read Coil (Output) Status, 188
    Código de función 01 - Read Coil (Output) Status:Acceso a marcas y salidas, 188
    Código de función 01 - Read Coil (Output) Status:Acceso a temporizadores y contadores, 189
    Código de función 01 - Read Coil (Output) Status:bit_number, 189
    Código de función 01 - Read Coil (Output) Status:Ejemplo de aplicación, 189
    Código de función 01 - Read Coil (Output) Status:start_address, 188
    Código de función 02 – Read Input Status, 192
    Código de función 02 – Read Input Status:bit_number, 192
    Código de función 02 – Read Input Status:Ejemplo de aplicación, 193
    Código de función 02 – Read Input Status:start_address, 192
    Código de función 03 -- Read Output Registers, 195
    Código de función 03 – Read Output Registers:Ejemplo de aplicación, 196
    Código de función 03 – Read Output Registers:Fórmula de cálculo para start_register, 195
    Código de función 03 – Read Output Registers:register_number, 196
    Código de función 03 – Read Output Registers:start_adress, 195
    -Código de función 04 -- Read Input Registers, 198
    Código de función 04 – Read Input Registers:register_number, 199
    Código de función 04 – Read Input Registers:start_address, 198
    Código de función 04 – Read Output Registers:Ejemplo de aplicación, 199
    Código de función 05 -- Force Single Coil, 201
    Código de función 05 – Force Single Coil:Acceso a marcas y salidas, 201
    Código de función 05 – Force Single Coil:Acceso a temporizadores y contadores, 202
    Código de función 05 -- Force Single Coil:coil_address, 201
    Código de función 05 – Force Single Coil:DATA-on/off, 202
    Código de función 05 – Force Single Coil:Ejemplo de aplicación, 202
    Código de función 06 -- Preset Single Register, 204
    Código de función 06 – Preset Single Register:DATA Value, 205
    Código de función 06 – Preset Single Register:Ejemplo de aplicación, 205
    Código de función 06 – Preset Single Register:start_register, 204
    Código de función 08 -- Loop Back Diagnostic Test, 207
    Código de función 08 - Loop Back Diagnostic Test:Ejemplo de aplicación, 207
    Código de función 15 -- Force Multiple Coils, 208
    Código de función 15 – Force Multiple Coils: Ejemplo de aplicación de la conversión de la asignación de dirección Modbus, 209
    Código de función 15 – Force Multiple Coils:Acceso a marcas y salidas, 208

Código de función 15 – Force Multiple Coils:Acceso a temporizadores y contadores, 209
Código de función 15 – Force Multiple Coils:DATA, 209
Código de función 15 – Force Multiple Coils:Quantity, 209
Código de función 15 -- Force Multiple Coils:start_address, 208
Código de función 16 – Preset Multiple Coils:Quantity, 212
Código de función 16 -- Preset Multiple Registers, 211
Código de función 16 – Preset Multiple Registers:DATA (High, Low), 212
Código de función 16 – Preset Multiple Registers:Ejemplo de aplicación, 212
Código de función 16 -- Preset Multiple Registers:start_register, 211
Códigos de función de maestro, 144
Código de función de maestro 01 -- Read Exception Status, 170
Código de función de maestro 01 -- Read Output Status, 164
Código de función de maestro 01 -- Read Output Status:DB de destino RCV, 164
Código de función de maestro 01 -- Read Output Status:DB de origen SEND, 164
Código de función de maestro 02 -- Read Input Status, 165
Código de función de maestro 02 -- Read Input Status:DB de destino RCV, 165
Código de función de maestro 02 -- Read Input Status:DB de origen SEND, 165
Código de función de maestro 03 -- Read Output Registers, 166
Código de función de maestro 03 -- Read Output Registers:DB de destino RCV, 166
Código de función de maestro 03 -- Read Output Registers:DB de origen SEND, 166
Código de función de maestro 04 -- Read Input Registers, 167
Código de función de maestro 04 -- Read Input Registers:DB de destino RCV, 167
Código de función de maestro 04 -- Read Input Registers:DB de origen SEND, 167
Código de función de maestro 05 -- Force Single Coil, 168
Código de función de maestro 06 -- Preset Multiple Registers, 175
Código de función de maestro 06 -- Preset Single Register, 169
Código de función de maestro 06 -- Preset Single Register:DB de origen SEND, 169
Código de función de maestro 07 -- Read Exception Status:DB de destino RCV, 170
Código de función de maestro 07 -- Read Exception Status:DB de origen SEND, 170
Código de función de maestro 08 -- Loop Back Diagnostic Test, 171
Código de función de maestro 08 -- Loop Back Diagnostic Test:DB de destino RCV, 171
Código de función de maestro 08 -- Loop Back Diagnostic Test:DB de origen SEND, 171
Código de función de maestro 11 – Fetch Communications Event Counter, 172
Código de función de maestro 11 – Fetch Communications Event Counter:DB de destino RCV, 172
Código de función de maestro 11 – Fetch Communications Event Counter:DB de origen SEND, 172
Código de función de maestro 12 -- Fetch Communications Event Log, 173
Código de función de maestro 12 – Fetch Communications Event Log:DB de destino RCV, 173
Código de función de maestro 12 – Fetch Communications Event Log:DB de origen SEND, 173
Código de función de maestro 15 - Force Multiple Coils, 174
Código de función de maestro 15 - Force Multiple Coils:DB de origen SEND, 174
Código de función de maestro 15 - Force Single Coil:DB de destino RCV, 168
Código de función de maestro 15 - Force Single Coil:DB de origen SEND, 168
Código de función del maestro 06 -- Preset Single Register:DB de destino RCV, 169
Códigos de función de maestro 16 -- Preset Multiple Registers:DB de origen SEND, 175
Comunicación RS 232C
Asignación de pines, 132
Comunicación RS-422
Asignación de pines, 133
Comunicación RS-485
Asignación de pines, 133
Configuración del módulo Modbus, 158, 185

D

Datos técnicos
Módulo interfaz serie, 117
Protocolos e interfaces, 273
DB de destino RCV
Código de función de maestro 01, 164
Código de función de maestro 02, 165
Código de función de maestro 03, 166
Código de función de maestro 04, 167
Código de función de maestro 05, 168
Código de función de maestro 06, 169
Código de función de maestro 07, 170
Código de función de maestro 08, 171
Código de función de maestro 11, 172
Código de función de maestro 12, 173
DB de origen SEND
Código de función de maestro 01, 164
Código de función de maestro 02, 165
Código de función de maestro 03, 166
Código de función de maestro 04, 167
Código de función de maestro 05, 168
Código de función de maestro 06, 169
Código de función de maestro 07, 170
Código de función de maestro 08, 171
Código de función de maestro 11, 172
Código de función de maestro 12, 173
Código de función de maestro 15, 174
Código de función de maestro 16, 175
Diagnóstico, 227
Clase de evento 14 (0E Hex) Drivers cargables – Error general de procesamiento <Parametrización>, 232
Clase de evento 14 (0E Hex) Drivers cargables – Errores generales de procesamiento <Procesamiento de una petición S_SEND>, 233
Clase de evento 30 (1EH): Error durante la comunicación entre SI y CPU, 237
Diagnóstico de esclavos PROFIBUS, 238
Estructura de los mensajes de diagnóstico de los bloques de función, 228
Información de diagnóstico de los LEDs de estado, 227
Llamada de la variable SFCERR, 228
Mensajes de diagnóstico de los bloques de función, 229
Tipos de error de canal del módulo interfaz serie ET 200S, 238
Diagnóstico del FB de comunicación
Borrado de errores, 239
Diagnóstico mediante los parámetros ERROR_NR, ERROR_INFO, 239
Diagnóstico mediante los parámetros ERROR_NR, ERROR_INFO:ERROR_No 1...9, 239
Diagnóstico mediante los parámetros ERROR_NR, ERROR_INFO:ERROR_No 10...19, 239
Diagnóstico mediante los parámetros ERROR_NR, ERROR_INFO:ERROR_No 90...99, 239
ERROR_NR, ERROR_INFO, 239
Errores durante la inicialización, 240
Funciones de diagnóstico, 239
Directrices de cableado, 132
Driver ASCII
Señales cualificadoras RS 232C, 51
Driver esclavo
Componentes:Acoplamiento de esclavo Modbus, 176
Componentes:Estructura de datos, 176
Componentes:FB de comunicación esclavo Modbus, 177
Parámetro:Bits de datos, 185
Parámetro:Bits de parada, 185
Parámetro:Dirección del esclavo, 186
Parámetro:Modo normal, 186
Parámetro:Multiplicador de retardo de caracteres, 186
Parámetro:Paridad, 186
Parámetro:Supresión de frecuencias perturbadoras, 186
Parámetro:Velocidad de transmisión, 185
Parámetros:Configurar, 182
Parámetros:Parametrización del driver esclavo, 182
Driver esclavo Modbus, 176

E

Ejemplo de aplicación de esclavo
Código de función 06 – Preset Single Register, 205
Código de función de esclavo 02, 193
Ejemplo sobre la puesta en servicio
Interfaz serie, 124
Ejemplos de aplicación de esclavo
Código de función 05, 202
Código de función 08, 207
Código de función 15:Conversión de la asignación de dirección Modbus, 209
Código de función 16, 212
Código de función de esclavo 01, 189
Código de función de esclavo 03, 196
Código de función de esclavo 04, 199

Esclavo
Dirección, 143
Esquemas de conexiones con asignación de terminales, 132
Estructura del telegrama, 142

F

FB2 S_RCV, 153
Asignación en el área de datos, 156
Cronograma, 157
FB3 S_SEND, 149
Asignación en el área de datos, 152
Cronograma, 153
Llamada, 151
Parámetro, 152
Fin del telegrama, 145

I

Interfaces
RS-232C, 138
RS-422/485, 141
Interfaces y protocolos aplicables, 148

M

Maestro USS, 243
DB de datos de red, 259
DB de datos de red:Asignación de datos de esclavo, 260
DB de datos de red:Byte PAFE 1, 264
DB de datos de red:Byte PAFE 2, 264
DB de datos de red:Estructura de la palabra de error de comunicación (DBW n+6), 263
DB de datos de red:ID de parámetro PKE en el búfer de envío, 264
DB de datos de red:Palabra de mando de comunicación (DBWn+4), 262
DB de datos de red:Palabra de mando de comunicación KSTW(DBWn), 261
DB de parametrización, 265
DB de parametrización:Parámetros de comunicación de esclavo, 266
DB de parametrización:Parámetros de sistema, 266
DB del procesador de comunicaciones, 267
DB del procesador de comunicaciones:Duración dle último ciclo DBW6, 268
DB del procesador de comunicaciones:Estado de comunicación DBW0, 268
FC17 S_USST: Enviar datos a un esclavo, 249
FC17 S_USST: Enviar datos a un esclavo:Parámetros, 251
FC18 S_USSR: Recibir datos de un esclavo, 252
FC18 S_USSR: Recibir datos de un esclavo:Parámetros, 255
FC19 S_USSI: Inicialización, 256
FC19 S_USSI: Inicialización:Parámetros, 258
Orden de las llamadas de funciones, 247
Protocolo USS, 244
Protocolo USS: Bloque de datos de red, 245
Protocolo USS:Codificación de datos, 244
Protocolo USS:Estructura de telegrama, 244
Protocolo USS:Procedimiento de transmisión de datos, 244
Sinopsis de funciones, 247
Modo de operación del acoplamiento para el driver esclavo
Acceso con códigos de función orientados a bits, 214
Acceso con códigos de función orientados a bits:Código de función 02, 214
Acceso con códigos de función orientados a registros, 215
Acceso con códigos de función orientados a registros:Código de función 04, 215
Acceso con códigos de función orientados a registros:Número de DB resultante, 215
Acceso con códigos de función orientados a registros:Número de palabra en el DB, 215
Área de datos en la CPU SIMATIC:Transformación de dirección, 180
Conversión de las direcciones MODBUS para funciones de bits, 217
Conversión de las direcciones MODBUS para funciones de bits:Área de memoria SIMATIC "A partir de",
Conversión de las direcciones MODBUS para funciones de bits:Dirección Modbus De/A, 218, 220
Conversión de las direcciones Modbus para funciones de bits:Ejemplo de FC 01, 05, 15, 219
Conversión de las direcciones Modbus para funciones de bits:Ejemplo de FC 02, 221
Conversión de las direcciones Modbus para funciones de bits:Sinopsis de FC 01, 05, 15, 217
Conversión de las direcciones MODBUS para funciones de bits:Sinopsis de FC02, 220
Conversión de las direcciones Modbus para funciones de registro:A partir de DB, 224
Conversión de las direcciones MODBUS para funciones de registro:Ejemplo, 223
Conversión de las direcciones Modbus para funciones de registros:Ejemplo de FC 04, 224

Conversión de las direcciones Modbus para funciones de registros:Sinopsis de FC 03, 06, 16, 222
Conversión de las direcciones MODBUS para funciones de registros:Sinopsis de FC 04, 223
Habilitar/bloquear accesos de escritura, 216
Habilitar/bloquear accesos de escritura:Códigos de función 05, 06,15, 16, 216
Límites de accesos de escritura:Ejemplo de FC 05, 06, 16, 226
Límites de accesos de escritura:Sinopsis de FC 05, 06, 15, 16, 225
Límites de las funciones de escritura:Áreas de memoria SIMATIC MIN/MAX, 225
Límites para accesos de escritura, 225
Modo dúplex, 26
Modo semidúplex, 26
Módulo interfaz serie
Datos técnicos, 117, 118, 274

P

Parametrizar el driver maestro, 158
Bits de datos, 161
Bits de parada, 161
Modo normal, 161
Multiplicador de retardo de caracteres, 162
Ocupación estándar de la línea de recepción, 160, 185
Paridad, 161
Semidúplex (RS485), modo a dos hilos, 160, 185
Supresión de frecuencias perturbadoras, 161
Tiempo de respuesta, 161
Velocidad de transmisión, 161
Procedimiento 3964(R), 35
Propiedades de arranque y estados operativos
Cargar los datos de parametrización, 269
Estados operativos, 270
Propiedades de arranque y estados operativos
Comportamiento del módulo ET 200S Modbus/USS cuando la CPU cambia de estado operativo, 271271
Propiedades del arranque, 271
Protocolo de transmisión, 142
Protocolo USS
Estructura general del bloque de datos de red:Área de datos de proceso (PZD), 245
Estructura general del bloque de datos de red:Área de parámetros (PKW), 245

R

Relación de productos
Número de referencia, 121

S

Señales
Cronograma de las señales cualificadoras, 140
Manejo automático de las señales cualificadoras, 139
RS-232C, 138
Señales cualificadoras RS 232C, 51

T

Telegrama de código de excepción, 146
Telegrama de transferencia, 143
Transferencia de datos con el maestro Modbus ET 200S, 149

V

Verificación CRC, 145